

MultiNet for OpenVMS User's Guide

Part Number: N-5010-43-NN-A

August 2000

This document describes how to use the MultiNet user commands. Included are easy to follow instructions for beginning users and command pages for advanced users.

Revision/Update: This manual supersedes the *MultiNet User's Guide*, Version V4.2

Operating System/Version: VAX/VMS V5.5-2 or later, OpenVMS VAX V6.0 or later, or OpenVMS Alpha V6.1 or later

Software Version: MultiNet V4.3

**Process Software
Framingham, Massachusetts
USA**

The material in this document is for informational purposes only and is subject to change without notice. It should not be construed as a commitment by Process Software. Process Software assumes no responsibility for any errors that may appear in this document.

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

The following third-party software may be included with your product and will be subject to the software license agreement.

Network Time Protocol (NTP). Copyright © 1992 by David L. Mills. The University of Delaware makes no representations about the suitability of this software for any purpose.

Point-to-Point Protocol. Copyright © 1989 by Carnegie-Mellon University. All rights reserved. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by Carnegie Mellon University. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

RES_RANDOM.C. Copyright © 1997 by Niels Provos <provos@physnet.uni-hamburg.de> All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by Niels Provos.
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Copyright © 1990 by John Robert LoVerso. All rights reserved. Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by John Robert LoVerso.

Kerberos. Copyright © 1989, DES.C and PCBC_ENCRYPT.C Copyright © 1985, 1986, 1987, 1988 by Massachusetts Institute of Technology. Export of this software from the United States of America is assumed to require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting. WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

DNSSIGNER (from BIND distribution) Portions Copyright (c) 1995-1998 by Trusted Information Systems, Inc. Portions Copyright (c) 1998-1999 Network Associates, Inc.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. THE

SOFTWARE IS PROVIDED "AS IS" AND TRUSTED INFORMATION SYSTEMS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL TRUSTED INFORMATION SYSTEMS BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

ERRWARN.C. Copyright © 1995 by RadioMail Corporation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of RadioMail Corporation, the Internet Software Consortium nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY RADIOMAIL CORPORATION, THE INTERNET SOFTWARE CONSORTIUM AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL RADIOMAIL CORPORATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software was written for RadioMail Corporation by Ted Lemon under a contract with Vixie Enterprises. Further modifications have been made for the Internet Software Consortium under a contract with Vixie Laboratories.

IMAP4R1.C, MISC.C, RFC822.C, SMTP.C Original version Copyright © 1988 by The Leland Stanford Junior University

NS_PARSER.C Copyright © 1984, 1989, 1990 by Bob Corbett and Richard Stallman

This program is free software. You can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 1, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139 USA

IF_ACP.C Copyright © 1985 and IF_DDA.C Copyright © 1986 by Advanced Computer Communications

IF_PPP.C Copyright © 1993 by Drew D. Perkins

ASCII_ADDR.C Copyright © 1994 Bell Communications Research, Inc. (Bellcore)

DEBUG.C Copyright © 1998 by Lou Bergandi. All Rights Reserved.

NTP_FILEGEN.C Copyright © 1992 by Rainer Pruy Friedrich-Alexander Universitaet Erlangen-Nuernberg

RANNY.C Copyright © 1988 by Rayan S. Zachariassen. All Rights Reserved.

MD5.C Copyright © 1990 by RSA Data Security, Inc. All Rights Reserved.

Portions Copyright © 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989 by SRI International

Portions Copyright © 1984, 1989 by Free Software Foundation

Portions Copyright © 1993, 1994, 1995, 1996, 1997, 1998 by the University of Washington. Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notices appear in all copies and that both the above copyright notices and this permission notice appear in supporting documentation, and that the name of the University of Washington or The Leland Stanford Junior University not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. This software is made available "as is", and THE UNIVERSITY OF WASHINGTON AND THE LELAND STANFORD JUNIOR UNIVERSITY DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, WITH REGARD TO THIS SOFTWARE, INCLUDING WITHOUT LIMITATION ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND IN NO EVENT SHALL THE UNIVERSITY OF WASHINGTON OR THE LELAND STANFORD JUNIOR UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, TORT (INCLUDING NEGLIGENCE) OR STRICT LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Portions Copyright © 1980, 1982, 1985, 1986, 1988, 1989, 1990, 1993 by The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright © 1993 by Compaq Computer Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Compaq Computer Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission. THE SOFTWARE IS PROVIDED "AS IS" AND COMPAQ COMPUTER CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL COMPAQ COMPUTER CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR

PERFORMANCE OF THIS SOFTWARE.

Portions Copyright © 1995 by International Business Machines, Inc.

International Business Machines, Inc. (hereinafter called IBM) grants permission under its copyrights to use, copy, modify, and distribute this Software with or without fee, provided that the above copyright notice and all paragraphs of this notice appear in all copies, and that the name of IBM not be used in connection with the marketing of any product incorporating the Software or modifications thereof, without specific, written prior permission. To the extent it has a right to do so, IBM grants an immunity from suit under its patents, if any, for the use, sale or manufacture of products to the extent that such products are used for performing Domain Name System dynamic updates in TCP/IP networks by means of the Software. No immunity is granted for any product per se or for any other function of any product. THE SOFTWARE IS PROVIDED "AS IS", AND IBM DISCLAIMS ALL WARRANTIES, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL IBM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE, EVEN IF IBM IS APPRISED OF THE POSSIBILITY OF SUCH DAMAGES.

Portions Copyright © 1995, 1996, 1997, 1998, 1999, 2000 by Internet Software Consortium. All Rights Reserved. Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1996-2000 Internet Software Consortium.

Use is subject to license terms which appear in the file named ISC-LICENSE that should have accompanied this file when you received it. If a file named ISC-LICENSE did not accompany this file, or you are not sure the one you have is correct, you may obtain an applicable copy of the license at: <http://www.isc.org/isc-license-1.0.html>. This file is part of the ISC DHCP distribution. The documentation associated with this file is listed in the file DOCUMENTATION, included in the top-level directory of this release. Support and other services are available for ISC products - see <http://www.isc.org> for more information.

ISC LICENSE, Version 1.0

1. This license covers any file containing a statement following its copyright message indicating that it is covered by this license. It also covers any text or binary file, executable, electronic or printed image that is derived from a file that is covered by this license, or is a modified version of a file covered by this license, whether such works exist now or in the future. Hereafter, such works will be referred to as "works covered by this license," or "covered works."
2. Each source file covered by this license contains a sequence of text starting with the copyright message and ending with "Support and other services are available for ISC products - see <http://www.isc.org> for more information." This will hereafter be referred to as the file's Bootstrap License.
3. If you take significant portions of any source file covered by this license and include those portions in some other file, then you must also copy the Bootstrap License into that other file, and that file becomes a covered file. You may make a good-faith judgement as to where in this file the bootstrap license should appear.
4. The acronym "ISC", when used in this license or generally in the context of works covered by this license, is an abbreviation for the words "Internet Software Consortium."
5. A distribution, as referred to hereafter, is any file, collection of printed text, CD ROM, boxed set, or other collection, physical or electronic, which can be distributed as a single object and which contains one or more works covered by this license.
6. You may make distributions containing covered files and provide copies of such distributions to whomever you choose, with or without charge, as long as you obey the other terms of this license. Except as stated in (9), you may include as many or as few covered files as you choose in such distributions.
7. When making copies of covered works to distribute to others, you must not remove or alter the Bootstrap License. You may not place your own copyright message, license, or similar statements in the file prior to the

original copyright message or anywhere within the Bootstrap License. Object files and executable files are exempt from the restrictions specified in this clause.

8. If the version of a covered source file as you received it, when compiled, would normally produce executable code that would print a copyright message followed by a message referring to an ISC web page or other ISC documentation, you may not modify the file in such a way that, when compiled, it no longer produces executable code to print such a message.

9. Any source file covered by this license will specify within the Bootstrap License the name of the ISC distribution from which it came, as well as a list of associated documentation files. The associated documentation for a binary file is the same as the associated documentation for the source file or files from which it was derived. Associated documentation files contain human-readable documentation which the ISC intends to accompany any distribution.

If you produce a distribution, then for every covered file in that distribution, you must include all of the associated documentation files for that file. You need only include one copy of each such documentation file in such distributions.

Absence of required documentation files from a distribution you receive or absence of the list of documentation files from a source file covered by this license does not excuse you from this from this requirement. If the distribution you receive does not contain these files, you must obtain them from the ISC and include them in any redistribution of any work covered by this license. For information on how to obtain required documentation not included with your distribution, see: <http://www.isc.org/getting-documentation.html>.

If the list of documentation files was removed from your copy of a covered work, you must obtain such a list from the ISC. The web page at <http://www.isc.org/getting-documentation.html> contains pointers to lists of files for each ISC distribution covered by this license.

It is permissible in a source or binary distribution containing covered works to include reformatted versions of the documentation files. It is also permissible to add to or modify the documentation files, as long as the formatting is similar in legibility, readability, font, and font size to other documentation in the derived product, as long as any sections labeled CONTRIBUTIONS in these files are unchanged except with respect to formatting, as long as the order in which the CONTRIBUTIONS section appears in these files is not changed, and as long as the manual page which describes how to contribute to the Internet Software Consortium (hereafter referred to as the Contributions Manual Page) is unchanged except with respect to formatting.

Documentation that has been translated into another natural language may be included in place of or in addition to the required documentation, so long as the CONTRIBUTIONS section and the Contributions Manual Page are either left in their original language or translated into the new language with such care and diligence as is required to preserve the original meaning.

10. You must include this license with any distribution that you make, in such a way that it is clearly associated with such covered works as are present in that distribution. In any electronic distribution, the license must be in a file called "ISC-LICENSE".

If you make a distribution that contains works from more than one ISC distribution, you may either include a copy of the ISC-LICENSE file that accompanied each such ISC distribution in such a way that works covered by each license are all clearly grouped with that license, or you may include the single copy of the ISC-LICENSE that has the highest version number of all the ISC-LICENSE files included with such distributions, in which case all covered works will be covered by that single license file. The version number of a license appears at the top of the file containing the text of that license, or if in printed form, at the top of the first page of that license.

11. If the list of associated documentation is in a separated file, you must include that file with any distribution you make, in such a way that the relationship between that file and the files that refer to it is clear. It is not permissible to merge such files in the event that you make a distribution including files from more than one ISC distribution, unless all the Bootstrap Licenses refer to files for their lists of associated documentation, and those references all list the same filename.

12. If a distribution that includes covered works includes a mechanism for automatically installing covered works, following that installation process must not cause the person following that process to violate this license, knowingly or unknowingly. In the event that the producer of a distribution containing covered files accidentally or wilfully violates this clause, persons other than the producer of such a distribution shall not be held liable for such violations, but are not otherwise excused from any requirement of this license.

13. COVERED WORKS ARE PROVIDED "AS IS". ISC DISCLAIMS ALL WARRANTIES WITH REGARD TO COVERED WORKS INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

14. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OF COVERED WORKS.

Use of covered works under different terms is prohibited unless you have first obtained a license from ISC granting use pursuant to different terms. Such terms may be negotiated by contacting ISC as follows:

Internet Software Consortium
950 Charter Street
Redwood City, CA 94063
Tel: 1-888-868-1001 (toll free in U.S.)
Tel: 1-650-779-7091
Fax: 1-650-779-7055
Email: info@isc.org
Email: licensing@isc.org

DNSSAFE LICENSE TERMS

This BIND software includes the DNSsafe software from RSA Data Security, Inc., which is copyrighted software that can only be distributed under the terms of this license agreement.

The DNSsafe software cannot be used or distributed separately from the BIND software. You only have the right to use it or distribute it as a bundled, integrated product.

The DNSsafe software can ONLY be used to provide authentication for resource records in the Domain Name System, as specified in RFC 2065 and successors. You cannot modify the BIND software to use the DNSsafe software for other purposes, or to make its cryptographic functions available to end-users for other uses.

If you modify the DNSsafe software itself, you cannot modify its documented API, and you must grant RSA Data Security the right to use, modify, and distribute your modifications, including the right to use any patents or other intellectual property that your modifications depend upon.

You must not remove, alter, or destroy any of RSA's copyright notices or license information. When distributing the software to the Federal Government, it must be licensed to them as "commercial computer software" protected under 48 CFR 12.212 of the FAR, or 48 CFR 227.7202.1 of the DFARS.

You must not violate United States export control laws by distributing the DNSsafe software or information about it, when such distribution is prohibited by law.

THE DNSSAFE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY WARRANTY WHATSOEVER. RSA HAS NO OBLIGATION TO SUPPORT, CORRECT, UPDATE OR MAINTAIN THE RSA SOFTWARE. RSA DISCLAIMS ALL WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO ANY MATTER WHATSOEVER, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS.

If you desire to use DNSsafe in ways that these terms do not permit, please contact:

RSA Data Security, Inc.
100 Marine Parkway
Redwood City, California 94065, USA
to discuss alternate licensing arrangements.

Secure Shell (SSH). Copyright © 2000. This License agreement, including the Exhibits ("Agreement"), effective as of the latter date of execution ("Effective Date"), is hereby made by and between Data Fellows, Inc., a California corporation, having principal offices at 675 N. First Street, 8th floor, San Jose, CA 95112170 ("Data Fellows") and Process Software, Inc., a Massachusetts corporation, having a place of business at 959 Concord Street, Framingham, MA 01701 ("OEM").

All other trademarks, service marks, registered trademarks, or registered service marks mentioned in this document are the property of their respective holders.

MultiNet is a registered trademark and Process Software and the Process Software logo are trademarks of Process Software.

Copyright ©1997, 1998, 1999, 2000 Process Software. All rights reserved. Printed in USA.

If the examples of URLs, domain names, internet addresses, and web sites we use in this documentation reflect any that actually exist, it is not intentional and should not to be considered an endorsement, approval, or recommendation of the actual site, or any products or services located at any such site by Process Software. Any resemblance or duplication is strictly coincidental.

Contents

Chapter 1 Introduction

Typographical Conventions	1-2
Further Reading	1-3

Chapter 2 Exploring Your Network Environment

Specifying Remote Hosts	2-1
Displaying Names of Other Users	2-1
Displaying Host Information	2-2
Displaying User Information	2-2
Interacting with Another User	2-4
Restrictions for Using TALK	2-5
Sending Reminders to Yourself	2-5

Chapter 3 Sending and Receiving Electronic Mail

Using OpenVMS Mail Across the Network	3-1
Specifying Addresses	3-1
Specifying a Host Alias	3-3
Specifying Individual Aliases	3-3
Using Mail Under ALL-IN-1	3-4

Chapter 4 Using Kerberos Authentication

Understanding Kerberos	4-1
Making Sure Kerberos is Available	4-2
Acquiring and Deleting Tickets	4-2
Obtaining Tickets Under Another User Name	4-3
Using Kerberos with the RCP, RLOGIN, RSHELL, and TELNET Commands	4-3
Checking Ticket Status	4-3

Changing Your Kerberos Password.....	4-3
--------------------------------------	-----

Chapter 5 Accessing Remote Systems with the RSHELL, RLOGIN, and TELNET Utilities

Executing Commands on a Remote System Using RSHELL	5-1
Using RSHELL	5-1
Interrupting and Terminating RSHELL	5-2
Logging Into a Remote System with RLOGIN.....	5-2
Using RLOGIN	5-2
Terminating an RLOGIN Session	5-3
"R" Services Authentication	5-3
Host Equivalences	5-3
User Equivalences	5-3
Cautions Concerning Use of Equivalences	5-4
Logging Into a Remote System with TELNET	5-5
Starting a TELNET Connection	5-5
Using TELNET Commands	5-5
Using TELNET Control Sequences	5-7
Running Applications over TELNET Connections	5-8
Accessing IBM Hosts with the TELNET Command	5-9
Starting TELNET with an IBM Terminal Emulator	5-9
Stopping an IBM Emulator Session	5-10
IBM 3278 Models	5-10
Mapping Your Keyboard	5-10
Displaying the Current Keyboard Mapping	5-10
Keyboard Mapping File Format	5-13
Functions	5-13
Specifying Multiple Keystrokes	5-13
TN3270 Function Key Mapping	5-14
TN5250 Function Key Mapping	5-16
Editing the Keyboard Mapping File	5-18
Capturing Screen Output and Printing Screen Captures	5-18
Using Transparent Mode	5-18
Application Keypad Access for TN3270 and TN5250	5-19
TN3270 Emulation	5-19
TN3270 Translation Table Mapping	5-19
Troubleshooting TELNET	5-21
Connection Problems	5-21
Problems Logging In	5-21

Chapter 6 Remote File Access with the RCP, FTP, and TFTP Utilities

Copying Files Using RCP	6-1
-------------------------------	-----

Requirements for RCP	6-1
Using RCP	6-2
Inhibiting Output from SYLOGIN.COM and LOGIN.COM	6-3
Accessing Files with FTP	6-3
Requirements for Using FTP	6-3
Invoking FTP and Logging In	6-3
Using FTP Commands	6-5
Getting FTP Command Help	6-5
Using Basic FTP Commands	6-6
Specifying TCP Window Size with FTP	6-6
File Name Translations	6-7
Listing the Contents of a File	6-10
Working with Directories	6-10
Commands for Copying Files	6-10
Parameters for Copying Files	6-11
FTP VMS Structure	6-11
FTP Commands While a Transfer is in Progress	6-12
Issuing FTP Commands From the DCL Command Line	6-12
FTP Command Scripts	6-13
Ending an FTP Session	6-13
FTP Log Files	6-14
Anonymous FTP	6-14
Transferring Files From Behind a Firewall	6-15
FTP Initialization File	6-15
Troubleshooting FTP	6-16
General Troubleshooting Tips	6-16
Transmitted Files Are Corrupt	6-17
Copying Files Using TFTP	6-17
Requirements for TFTP	6-17
Using TFTP	6-17

Chapter 7 Using DECwindows with MultiNet

Running DECwindows Applications	7-1
Authorizing Remote Systems	7-2

Chapter 8 Accessing Remote Systems with the Secure Shell (SSH) Utilities

Secure Shell Client (remote login program)	8-1
First authentication method	8-1
Second authentication method	8-1
Third authentication method	8-2
Fourth authentication method	8-4
Port Forwarding	8-8

- CONFIGURATION FILES 8-10
- Other Files 8-15
- SSHAgent (authentication agent) 8-20
 - DESCRIPTION 8-20
 - FILES 8-20
- SSHADD 8-21
 - DESCRIPTION 8-21
 - OPTIONS 8-21
 - RETURN STATUS 8-21
 - FILES 8-22
- SSHKEYGEN 8-22
 - DESCRIPTION 8-22
 - OPTIONS 8-23
 - FILES 8-24

Appendix A DCL User Commands

- Command Summary A-1
 - MULTINET DECODE A-3
 - MULTINET FINGER A-4
 - MULTINET FTP A-5
 - MULTINET KERBEROS DESTROY A-9
 - MULTINET KERBEROS INIT A-10
 - MULTINET KERBEROS LIST A-11
 - MULTINET KERBEROS PASSWORD A-12
 - MULTINET LPRM A-13
 - MULTINET RCP A-14
 - MULTINET REMIND A-18
 - MULTINET RLOGIN A-20
 - MULTINET RSHELL A-22
 - MULTINET RUSERS A-24
 - MULTINET SEND A-25
 - MULTINET TALK A-26
 - MULTINET TELNET A-28
 - MULTINET TFTP A-34
 - MULTINET WHOIS A-35

Appendix B FTP Command Reference

- FTP Command Summary B-1
 - ACCOUNT B-6
 - AGET B-7
 - APPEND GET B-8
 - APPEND PUT B-9

APPEND RECEIVE	B-10
APPEND SEND.....	B-11
APUT	B-12
ASCII	B-13
ATTACH	B-14
BELL.....	B-15
BINARY	B-16
BLOCK.....	B-17
BYE.....	B-18
BYTE	B-19
CD	B-20
CDUP.....	B-21
CLOSE.....	B-22
CONFIRM	B-23
CONNECT.....	B-24
CPATH	B-25
CREATE-DIRECTORY	B-26
CWD	B-27
DELETE	B-28
DIRECTORY.....	B-29
DISCONNECT	B-30
EXIT	B-31
EXIT-ON-ERROR.....	B-32
GET.....	B-33
HASH.....	B-34
HELP.....	B-35
LCD.....	B-36
LDIR	B-37
LIST	B-38
LOCAL-CD	B-39
LOCAL-DIRECTORY	B-40
LOCAL-PWD	B-41
LOGIN	B-42
LPWD	B-43
LS	B-44
MDELETE.....	B-45
MGET	B-46
MKDIR	B-47
MPUT	B-48
MULTIPLE DELETE.....	B-49
MULTIPLE GET	B-50
MULTIPLE PUT	B-51
MULTIPLE RECEIVE.....	B-52
MULTIPLE SEND	B-53
OPEN	B-54
PASSIVE	B-55

PASSWORD	B-57
PORT	B-58
PROMPT-FOR-MISSING-ARGUMENTS	B-59
PROMPT-ON-CONNECT	B-60
PUSH	B-61
PUT	B-62
PWD	B-64
QUIT	B-65
QUOTE	B-66
RECEIVE	B-67
RECORD-SIZE	B-68
REMOTE-HELP	B-69
REMOVE-DIRECTORY	B-70
RENAME	B-71
RETAIN	B-72
RM	B-73
RMDIR	B-74
SEND	B-75
SET	B-76
SHOW-DIRECTORY	B-77
SITE	B-78
SPAWN	B-79
STATISTICS	B-80
STATUS	B-81
STREAM	B-82
STRUCTURE	B-83
TAKE	B-84
TENEX	B-85
TYPE	B-86
USER	B-87
VERBOSE	B-88
VERSION	B-89

Appendix C TELNET Command Reference

Command Summary	C-1
ABORT	C-4
ATTACH	C-5
ATTN	C-6
AYT	C-7
BINARY	C-8
BREAK	C-9
BYE	C-10
CLOSE	C-11
CONNECT	C-12

CREATE-NTY	C-13
DEBUG	C-14
ECHO	C-15
EXIT	C-16
HELP	C-17
LOG-FILE	C-18
PUSH	C-19
QUIT	C-20
SET ABORT-OUTPUT-CHARACTER	C-21
SET ARE-YOU-THERE-CHARACTER	C-22
SET AUTO-FLUSH	C-23
SET BREAK-CHARACTER	C-24
SET DEBUG	C-25
SET ERASE-CHARACTER-CHARACTER	C-26
SET ERASE-LINE-CHARACTER	C-27
SET ESCAPE-CHARACTER	C-28
SET EXTENDED	C-29
SET INTERRUPT-PROCESS-CHARACTER	C-30
SET LOCAL-FLOW-CONTROL	C-31
SET LOG-FILE	C-32
SET REMOTE-USERNAME	C-33
SET UNIX-LINE-TERMINATOR	C-34
SPAWN	C-35
STATUS	C-36
TERMINAL-TYPE	C-37
VERSION	C-38

Appendix D TFTP Command Reference

Command Summary	D-1
CONNECT	D-2
GET	D-3
PUT	D-4
QUIT	D-5
REXMT	D-6
STATUS	D-7
TIMEOUT	D-8
TRACE	D-9

Index

MultiNet Master Index

Chapter 1

Introduction

This guide describes the commands (ones that do not require special privileges) and additional information and methods for using Process Software's MultiNet for OpenVMS.

This guide helps you with the following tasks:

To...	Read
Understand your network environment	Chapter 2, "Exploring Your Network Environment."
Send and receive e-mail	Chapter 3, "Sending and Receiving Electronic Mail."
Acquire and release Kerberos authentication tickets for use with the RCP, RLOGIN, RSHELL, and TELNET commands	Chapter 4, "Using Kerberos Authentication."
Log into a remote system	Chapter 5, "Accessing Remote Systems with the RSHELL, RLOGIN, and TELNET Utilities."
Transfer files to or from a remote system	Chapter 6, "Remote File Access with the RCP, FTP, and TFTP Utilities."
Use DECwindows with MultiNet	Chapter 7, "Using DECwindows with MultiNet."
Use Secure Shell (SSH)	Chapter 8, "Accessing Remote Systems with the Secure Shell (SSH) Utilities"

Reference material in the appendices provides more specific information about DCL, FTP, TELNET, and TFTP commands and their qualifiers:

Appendix	Topic
Appendix A	DCL commands you enter on the OpenVMS command line
Appendix B	Commands for transferring files between systems with FTP
Appendix C	Commands for logging into another system with TELNET
Appendix D	Commands for transferring files between systems with TFTP

Typographical Conventions

Examples in this guide use the following conventions:

Convention	Example	Meaning
Bold text	YES	Represents user input in instructions or examples.
Bold, uppercase Courier text	RETURN	Represents a key on your keyboard.
Bold Courier text with a slash	Ctrl/A	Indicates that you hold down the key labeled Control or Ctrl while simultaneously pressing another key; in this example, the "A" key.
A vertical bar within braces	{ ON OFF }	Indicates a list of values permitted in commands. The vertical bar separates alternatives; do not type the vertical bar in the actual command.
Italicized text	<i>file_name</i>	Represents a variable or placeholder; introduces new terminology or concepts; emphasizes something important; represents the title of a book or publication.
Square brackets	[FULL]	Indicates optional choices; you can enter none of the choices, or as many as you like. When shown as part of an example, square brackets are actual characters you should type.
Underscore or hyphen	<i>file_name</i> or <i>file- name</i>	Between words in commands, indicates the item is a single element.

Further Reading

For additional information on networking and TCP/IP architecture and management, enter this command to display information about some useful books and other documentation:

```
$ HELP MULTINET BOOKS
```


Chapter 2

Exploring Your Network Environment

This chapter helps you start exploring your network environment and covers the following topics:

- Specifying a remote host to contact
- Determining who is logged into your system or cluster or another site (using RUSERS)
- Displaying information registered by the Network Information Center (NIC) about your site or another site (using WHOIS)
- Displaying information about users, domains, hosts, and IP addresses (using FINGER)
- Contacting other users over the network (using TALK)
- Posting and receiving reminder messages (using REMIND)

Appendix A provides complete descriptions of the commands introduced in this chapter.

Specifying Remote Hosts

Most MultiNet applications allow you to specify a remote host by either name or Internet address. To access a host by name, the remote host must either be listed in the local system's host database or registered with a DNS (Domain Name System) server accessible from the local system. If you have difficulty accessing a remote host by its host name, contact your system manager or network administrator.

Displaying Names of Other Users

You can display a list of users on your system or on a remote system with the RUSERS command. For example:

```
$ MULTINET RUSERS
SURETE      RICK PATRICK
MIFIVE      MATT MATT MATT MATT
KGB         KEN KEN GIGI KEN JOEL JOEL JOEL JOEL
SCIENCE     RICK RICK RICK RICK
WHO         PATRICK PATRICK PATRICK PATRICK PATRICK PATRICK ROB ROB
DESIGN      BRUCE BRUCE BRUCE BRUCE BRUCE
```

CHAZ

GEORGE GEORGE GEORGE RICK RICK RICK GEORGE GEORGE GEORGE

The RUSERS utility uses the RUSERS Remote Procedure Call (RPC) service to display information about users logged into the local system or a remote system. It can display information about a particular system, or, if supported by the network hardware, use broadcasts to display information about all remote systems on directly connected networks. RUSERS uses UDP/IP (User Datagram Protocol/Internet Protocol) as the transport mechanism for the RPC services it calls. When using RUSERS, the command can appear to hang, but is in fact waiting for a timeout period to ensure that the last packet is received.

Note! If the system you are querying does not support the RUSERS RPC service, you will not receive any response (the RPC call times out silently).

Displaying Host Information

Use the WHOIS command to display information about a user, host, or domain accessed from the Internet's repository of information. The WHOIS command sends your request across the Internet to the NIC (at the RS.INTERNIC.NET host) and displays the information returned.

For example:

```
$ WHOIS ULANOV
Ulanov, V.I.          ulanov@abc.COM
ABC, Incorporated
100 Nevsky Street
Anytown, CA 95060
(408) 555-1212
Record last updated on 31-May-00.
```

The InterNIC Registration Services Host contains only Internet information (Networks, ASN's, Domains, and POC's).

```
$
```

Because RS.INTERNIC.NET is heavily used, you may receive a message stating that "the network is busy, try later." As an alternative, you can ask your system manager about possibly selecting another WHOIS server.

Displaying User Information

You can display information about a domain, host, IP address, or single user. The FINGER utility accesses information on your local system or on a remote system.

You can display information about your host, as shown in the following example:

```
$ MULTINET FINGER/NOCLUSTER
Monday, March 13, 2000 7:59PM-EST Up 1 10:33:01
nn+0 Jobs on CHUCKO Load ave 0.02 0.01 0.02
```

User	Personal Name	Job	Subsys	Terminal	Console	Location
BROWN	John Brown	40A0022C	MM	6.FTA13		

```

40A0022D EMACS 1:20.FTA14
40A0022E *DCL* 22.FTA15
40A0025F *DCL* 3:46.FTA23
40A00260 *DCL* 3:33.FTA24
40A00261 FINGER .FTA25
SYSTEM System Manager 23000120 *DCL* BIRD$RTA1 KARLA::PIPER
23000121 *DCL* BIRD$RTA2 KARLA::PIPER

```

If you want to display FINGER information about every node in a VMScluster, omit the /NOCLUSTER qualifier. To display information about another host, add its name to the end of the command:

\$ MULTINET FINGER

```

Monday, March 13, 2000 7:59PM-EST Up 1 10:33:01
nn+0 Jobs on CHUCKO Load ave 0.02 0.01 0.02

```

User	Personal Name	Job	Subsys	Terminal	Console Location
BROWN	John Brown	40A0022C	MM	6.FTA13	
		40A0022D	EMACS	1:20.FTA14	
		40A0022E	*DCL*	22.FTA15	
		40A0025F	*DCL*	3:46.FTA23	
		40A00260	*DCL*	3:33.FTA24	
		40A00261	FINGER	.FTA25	
SYSTEM	System Manager	23000120	*DCL*	BIRD\$RTA1	KARLA::PIPER
		23000121	*DCL*	BIRD\$RTA2	KARLA::PIPER
RICH	I. M. Rich	23200227	*DCL*	CODEZ\$NTY1	Rich.ABC.COM
POOR	U. R. Poor	2280027B	*DCL*	4\$FTA4	
JONES	Mary Jones	21C00C04	EMACS	SYS1\$NTY5	BigBird.ABC.COM

You can only display information about another system if a FINGER server is running there and if the system permits it (some do not). The information you receive can vary depending on the FINGER server in use.

To display information about users at a specific IP address, use this command format:

\$ MULTINET FINGER @192.192.192.1

```

Monday, March 13, 2000 7:59PM-EST Up 1 10:33:01
nn+0 Jobs on CHUCKO Load ave 0.02 0.01 0.02

```

User	Personal Name	Job	Subsys	Terminal	Console Location
BROWN	John Brown	40A0022C	MM	6.FTA13	
		40A0022D	EMACS	1:20.FTA14	
		40A0022E	*DCL*	22.FTA15	
		40A0025F	*DCL*	3:46.FTA23	
		40A00260	*DCL*	3:33.FTA24	
		40A00261	FINGER	.FTA25	
SYSTEM	System Manager	23000120	*DCL*	BIRD\$RTA1	KARLA::PIPER
		23000121	*DCL*	BIRD\$RTA2	KARLA::PIPER

The load average information displayed at the beginning of the FINGER output is the average

number of processes waiting for the CPU for the last one, two, and five minutes. For more information, ask your system manager.

To display information about a single user, use this command format:

```
$ MULTINET FINGER BROWN
BROWN      John      40A0022C MM      11.FTA13
              40A0022D EMACS      .FTA14
              40A0022E *DCL*      27.FTA15
              40A0025F FINGER      .FTA23
              40A00260 *DCL*      3:39.FTA24
              40A00261 *DCL*      2.FTA25

Mail from firefly@marx.edu (Rufus T. Firefly) at Mon 13-Mar-2000 7:53 PM-
EST Last read on Mon 13-Mar-2000 7:59 PM-EST
Plan: At the beach today. The higher, the fewer!
-- Alexander in the colony of free spirits (ST-TNG)
```

If you want specific information to be available when someone seeks information about you with FINGER, create a PLAN.TXT text file in your login directory. If you want to have a plan file on a UNIX system, create a .plan file in your login directory.

The information in this file is available even when you are not logged in. When you create this file, ensure the file has world read access (W:R) and your login directory has world execute permissions (W:E). You can insert any text (except control characters which are filtered out), and the file can be any length you want.

- If you FINGER a single user on a VMS system running MultiNet, the utility looks for a file named PLAN.TXT in that user's login directory. If that file does not exist, it looks for a file named .PLAN.
- If you FINGER a single user on a UNIX system, FINGER looks for a file named .plan.

Interacting with Another User

You can communicate with another user over the network using the TALK utility. TALK is similar to the OpenVMS PHONE utility except TALK can work with some non-OpenVMS operating systems.

TALK divides the screen into two sections; it displays text you enter in one section, and text entered by the other user in the other. You can then converse with each other until one of you presses **Ctrl/C** to end the session.

Use the following keystrokes during a TALK session:

Press..	To...	Press...	To...
Delete	Delete the last character typed	Ctrl/L	Redraw the screen
Ctrl/C	Exit and return to DCL command mode	Ctrl/W	Delete the last word typed

Restrictions for Using TALK

Some restrictions apply when using TALK:

- You and the person with whom you wish to TALK need to be on systems with the same byte-ordering scheme (either "Big Endian" or "Little Endian").

For example, if the other person is using a Sun workstation or a terminal connected to one, they cannot use the TALK command. Sun users need to use the NTALK command. NTALK is provided on the MultiNet software distribution CD-ROM in the [CONTRIBUTED-SOFTWARE.APPLICATIONS.NTALK] directory, or elsewhere as public domain software. Your system manager can provide more information.

- Both of your terminals must be able to accept broadcasts. Use these commands to enable broadcasts but suppress mail broadcasts:

```
$ SET TERMINAL /BROADCAST
$ SET BROADCAST=NOMAIL
```

- Your terminal type must be listed in the OpenVMS TERMTABLE.TXT database. As shipped with OpenVMS, this database includes all Compaq VT-series terminals. If you have a non-Compaq terminal, check with your system manager.
- The other person's system must be known to your system. TALK must be able to translate the remote system's IP address into its name. Your system must be using the Domain Name System (DNS) or have the remote system recorded in its host tables.

When a user uses TALK to call you, a message of the following form appears on your terminal:

```
Message from TALK-DAEMON@FLOWERS.COM at 1:53PM-PDT
Connection request by username
[Respond with: TALK username@hostname]
Type a TALK command to start the conversation:
$ TALK username@hostname
```

Once communication is established, you and the other user can type simultaneously, with your output appearing in separate windows.

If you try to TALK with a user who has disabled reception of broadcast messages, this message appears:

```
[Your party is refusing messages]
```

The TALK Server uses the PHONE operator class.

Note! To prevent users from attempting to TALK with you, use the SET BROADCAST=NOPHONE command.

Sending Reminders to Yourself

You can send reminders with the REMIND utility, as shown in the following example:

```
$ REMIND
REMIND Version V4.3(nn), 13-MAR-2000
There are no reminders in your remind file.
REMIND>CREATE
Time of first reminder? 22:45
Expiration count? 1
How should I send it? SEND
Addresses? ME
Subject? Testing
Text (end with ^Z)
This is a test.
^Z
REMIND>exit
[Entering your changes...]
```

When REMIND starts, it checks to see if any reminders are pending. It then displays the REMIND> prompt. Use the CREATE command to start a new reminder. The time of the reminder can be in 12-hour or 24-hour time and can also be a special name. The expiration count is the number of times you want the message sent. You can specify that the message be sent by mail, broadcast to the terminal ("send"), or both. You can enter details in much the same way as a mail message with the address of the recipient, the subject, and the text. When you press **Ctrl/Z**, the message is queued.

If you request reminders by mail, the information you specify is used to construct an electronic mail message. If you request reminders by broadcast to the terminal, REMIND sends a message like the following:

```
[REMIND(10:50PM): subject Message text]
```

For help, enter a question mark (?) at any prompt. For example, at the "Time of first reminder?" prompt, the following help appears:

```
Time of first reminder? ? date and time or one of the following:
FRIDAY MONDAY SATURDAY SUNDAY THURSDAY TODAY TOMORROW TUESDAY WEDNESDAY
or one of the following:
APRIL-FOOLS          BASTILLE-DAY          BEETHOVENS-BIRTHDAY
BLBOS-BIRTHDAY       CHRISTMAS              COLUMBUS-DAY
FLAG-DAY             FRODO'S-BIRTHDAY      GONDORIAN-NEW-YEAR
GROUND-HOG-DAY       GUY-FAWKES-DAY        HALLOWEEN
INDEPENDENCE-DAY     LEAP-DAY              LINCOLNS-BIRTHDAY
MAY-DAY              MEMORIAL-DAY          NEW-YEARS
SAINT-PATRICKS-DAY   SHERLOCK-HOLMES-BIRTHDAY MOZARTS-BIRTHDAY
```

Chapter 3

Sending and Receiving Electronic Mail

This chapter describes how to use OpenVMS MAIL and ALL-IN-1 Mail with MultiNet and covers the following major topics:

- Using OpenVMS mail across the network
- Using mail under ALL-IN-1 across the network

Using OpenVMS Mail Across the Network

MultiNet enhances OpenVMS Mail so you can send and receive mail across the network.

Specifying Addresses

When you use OpenVMS Mail to send mail to a host outside your VMScluster, the message is sent via SMTP (Simple Mail Transfer Protocol). For this reason, you must specify the address so that SMTP accepts the mail correctly. The format for the address is:

To: **SMTP%***recipient@destination*

The string SMTP and the destination system name are not case-sensitive; that is, you can type them in either uppercase or lowercase letters. The destination recipient specification may be case-sensitive, however, depending on the destination system's software. On some UNIX systems, ROOT and root specify two different user names (and hence different electronic mail addresses).

If the address contains an apostrophe, enter the address with either \' or \s as shown in the following example formats:

```
To: SMTP%"Thomas.O\'Malley@alley.cat.net"
To: SMTP%"Thomas.O\sMalley@alley.cat.net"
```

For the address <Thomas.O'Malley@alley.cat.net>.

To: **SMTP%***\'recipient@destination*

or

To: **SMTP%**"\srecipient@destination"

If the address is on a local DECnet network, use this format:

To: **SMTP%**nodename::username

If the address is on a remote DECnet network, you may use this format:

To: **SMTP%**"'nodename::username'@destination"

Note! MultiNet assumes that an address containing a double colon (::) is a DECnet address. If an address contains a double colon and is not a DECnet address, SMTP does not handle it correctly.

If you know the recipient's IP address, but not the host name (or if the host name is not registered in the Domain Name System), specify the recipient address as follows:

To: **smtp%**"recipient@[aa.bb.cc.dd]"

aa.bb.cc.dd is the destination system's IP address in dotted-decimal form. You must specify the IP address in square brackets.

The OpenVMS Mail utility also allows you to specify an addressee on the command line:

\$ **MAIL** filename addressee

To use this form of the command with MultiNet, you must enclose the address in quotes (and you must double all existing quotes), as follows:

\$ **MAIL** filename **smtp%** "recipient@destination"

The following example shows the user sending mail using the OpenVMS MAIL utility to a user named John Smith with a user name of "johns" on system SALES.FLOWERS.COM.

```
$ MAIL
MAIL>SEND
To:      SMTP%"johns@sales.flowers.com"
Subj:    This is a test message.
Enter your message below. Press Ctrl/Z when complete, or
Ctrl/C to quit:
Hi John, this is a test of the MultiNet extension to the VMS MAIL utility.
Ctrl/Z
MAIL>EXIT
$
```

You receive network mail as you would all other mail in the VMS MAIL utility. The following example shows the user "WHORFIN" reading an SMTP mail message sent by the user "johns."

```
$
New mail on node KAOS from SMTP%"johns@sales.flowers.com" "John Smith"
$ MAIL
You have 1 new message.
MAIL>READ/NEW
```

```
#1          03-13-2000 10:05:40.79
From:      SMTP%"johns@sales.flowers.com"      "John Smith"
To:        WHORFIN
CC:
Subj:      Re: This is a test message.
Date:      Mon, 13 Mar 2000 10:04:50 EST
From:      johns@sales.flowers.com (John Smith)
Message-Id: <891120100450.77@SALES.FLOWERS.COM>
Subject:    Re: This is a test message.
To:        whorfin@flowers.com
X-Vmsmail-To: SMTP%"whorfin@flowers.com"
Glad to see your test worked.
This is my response.
MAIL>EXIT
```

Specifying a Host Alias

MultiNet allows a system to have multiple names-or host aliases-with respect to electronic mail delivery. You can specify the host alias you want to use by defining the MULTINET SMTP FROM_HOST logical name. The alias you choose must be one of the SMTP host name aliases registered on the system (see the translation of the logical name MULTINET SMTP_HOST_NAME and the contents of the file MULTINET_HOST_ALIAS_FILE). If the alias you use is unknown, the setting of MULTINET SMTP FROM_HOST is ignored.

The host alias feature allows users from different administrative units within an organization to have their return address reflect the name of their unit, even though mail for all units is handled by one system.

Specifying Individual Aliases

MultiNet supports both *system-wide* and *per-user* mail aliases. Using these aliases, you can refer to electronic mail addresses with names that are meaningful to you. Per-user mail aliases are kept in the file SMTP_ALIASES. in your login directory.

The format for alias entries is:

```
alias:      real_address[,...];
```

alias is an alphanumeric string and *real_address* is an electronic mail address. You can specify multiple addresses by separating them with commas (.). The alias definition may span multiple lines, if needed, and must always be terminated with a semicolon (;).

For example, a local user may have a user name of JB134A, but you want to send mail to him as john. Add the following line to your SMTP_ALIASES. file:

```
john:      jbl34A;
```

Aliases are repeatedly translated until no more translations are found. You can circumvent the repeated translations by including a leading underscore (_) in the *real_address*. For example, this definition causes mail to be forwarded and delivered locally:

fnord: fnord@somewhere.else.edu, _fnord:

Using Mail Under ALL-IN-1

This section explains how to use the mail subsystem under ALL-IN-1 to send mail to and receive mail from users on remote systems.

To send mail to a user on a remote system, specify an ALL-IN-1 e-mail address in the format:

recipient@destination@SMTP

@SMTP indicates to the ALL-IN-1 mail subsystem that the message should be given to the SMTP/MR gateway facility for eventual handling by the MultiNet SMTP mail system.

Note! The string SMTP and the destination system name are not case-sensitive; that is, you can type them in either uppercase or lowercase letters. However, the destination recipient specification may be case-sensitive, depending on the destination system's software. On some UNIX systems, ROOT and root specify two different user names (and hence different electronic mail addresses).

You receive network mail as you would all other mail in the ALL-IN-1 mail subsystem. Contact your system manager for the correct syntax for remote users; frequently, the proper syntax is:

yourname@A1.yourdomain

Chapter 4

Using Kerberos Authentication

This chapter explains how to use the Kerberos authentication system, and covers the following topics:

- Kerberos principles
- Making sure Kerberos is available on your system
- Acquiring and deleting Kerberos tickets
- Using Kerberos with the RCP, RLOGIN, RSHELL, and TELNET commands
- Checking the status of tickets
- Changing your Kerberos password

Understanding Kerberos

Kerberos provides a secure way of proving a user's identity across an unsecure network. It does this without transmitting passwords where an intruder could see them. MultiNet has several enhanced or *Kerberized* commands including RCP, RLOGIN, RSHELL, and TELNET.

The process of proving one's identity is called *authentication*. Deciding whether or not to allow access to a resource is called *authorization*. Kerberos is an authentication system. Because authentication is a prerequisite to authorization, an application can make an authorization decision (for example, deciding to permit you to log in) based on your identity as authenticated by Kerberos.

Kerberos maintains a list of users and their encrypted passwords. Before you can use Kerberized commands, your system manager must have added your name to this list. You can only use Kerberized commands if you have a ticket for the command you wish to use. Analogous to the tickets you purchase when you go to a movie, Kerberos tickets permit you to invoke Kerberized utilities while you are logged in.

To use Kerberos, you must first:

- Acquire an initial ticket when you log in. This initial ticket, known as a *ticket-getting ticket* (or TGT), enables you to automatically get other tickets you will need to access application servers. You may also need to acquire another TGT when a previous one expires.

- Delete tickets before you log out. *It is very important to remember to delete your tickets any time you leave your terminal!* If another user "borrows" your tickets, you can be locked out of the network or impersonated by the intruder.
- Always run Kerberized utilities with the /AUTH qualifier. (The full form of the qualifier is /AUTHENTICATION=KERBEROS.)
- Change your Kerberos password at least once a month.

Kerberos security helps protect you and other users from data theft and other possible security breaches. You are the ultimate security element in making sure your files are safe; it is up to you to choose a password that is not easily guessed, and delete your tickets before you log out.

Making Sure Kerberos is Available

Before continuing with this chapter, make sure Kerberos is available on your system by asking your system manager these questions:

1	Is Kerberos enabled?
2	Has a Kerberos principal been created for me?
3	<div>Do I need to get and delete Kerberos tickets?<ul style="list-style-type: none">• If the answer to all three questions is yes, read this chapter.• If Kerberos is not enabled, skip to the next chapter.• If no Kerberos principal exists, your system manager must add one for you before you can use Kerberos.• If you answered no only to question 3, and yes to questions 1 and 2, you only need to read the section on changing your Kerberos password for information on changing your Kerberos password. All other commands are handled automatically on your system.</div>

Acquiring and Deleting Tickets

To acquire your initial ticket-getting ticket, enter this command from the DCL command line:

```
$ MULTINET KERBEROS INIT
This node is: holmes.flowers.com
Kerberos Initialization for "john"
Password: password
```

If you need to be authenticated as another user, use the /USERNAME qualifier. Use the /REALM qualifier to be authenticated in another realm. (A *realm* is an administrative name for a site, system, or other organizational entity.)

You can delete tickets with this command:

```
$ MULTINET KERBEROS DESTROY
```


Obtaining Tickets Under Another User Name

You can use the MULTINET KERBEROS INIT command with the /USERNAME qualifier to obtain tickets under another user name. For example, if you gained access to the system through a GUEST login, but you want to continue access to the network as yourself, you could use the /USERNAME qualifier with the MULTINET KERBEROS INIT command to specify your own user name. When you issue this form of the command, you are prompted for the other user's Kerberos password.

To access a remote system as another user, use both the /AUTH and /USERNAME qualifiers with the RCP, RLOGIN, RSHELL, and TELNET commands.

Using Kerberos with the RCP, RLOGIN, RSHELL, and TELNET Commands

The RCP, RLOGIN, RSHELL, and TELNET commands all support the /AUTHENTICATION=KERBEROS qualifier (specify this qualifier first before any other qualifiers). You can shorten this qualifier to /AUTH. For example:

```
$ RLOGIN/AUTH FLOWERS.COM
```

You can use the /USERNAME qualifier with the /AUTH qualifier to specify the user name you want to use to log into the remote system.

Checking Ticket Status

You can check the status of your tickets with the MULTINET KERBEROS LIST utility. For example, to test the status from the command line, enter:

```
$ MULTINET KERBEROS LIST
Principal:      john@FLOWERS.COM
Issued          Expires          Principal
June 13 16:16:47 June 14 00:16:47 krbgt.TROIKA.FOO@TROIKA.FOO
$
```

The utility also provides the /CHECK_TGT qualifier so you can test whether your ticket-getting ticket has already expired. If the ticket has expired, run MULTINET KERBEROS INIT again. The following command procedure tests your ticket status:

```
$! Test ticket status
$!
$ MULTINET KERBEROS LIST /CHECK_TGT
$ IF $STATUS THEN WRITE SYS$OUTPUT "Okay"
```

If the tickets are valid, \$STATUS is true. If the tickets have expired, \$STATUS is false.

Changing Your Kerberos Password

You can change your Kerberos password with this command:

```
$ MULTINET KERBEROS PASSWORD
Old password for holmes: password
```

```
New password for holmes: password
Verifying, re-enter New password for holmes: password
$
```

Use these guidelines for selecting a Kerberos user password:

- Kerberos passwords are case-sensitive so if you press the **SHIFT** key when you create the password, you must always press the key at the same point when entering the password.
- Kerberos passwords can be up to 64 characters long.
- Spaces and control characters are not permitted. In addition, you cannot use the **DELETE** key to correct a misspelling when entering a password.
- Select a password that is not a name, proper noun, and preferably not a common word. Intersperse letters and numbers in the string.

Chapter 5

Accessing Remote Systems with the RSH, RLOGIN, and TELNET Utilities

This chapter describes how to execute commands on remote systems using the RSH utility, and how to log into remote systems using the RLOGIN and TELNET utilities. The chapter covers the following topics:

- Executing commands on a remote system using the RSH utility
- Logging into a remote system using the RLOGIN utility
- Logging into a remote system using the TELNET utility

Executing Commands on a Remote System Using RSH

The RSH utility lets you execute commands on remote hosts. RSH connects to the specified host and creates an RSH server process to execute the commands you enter. If the remote command requires input, data is read from `SY$INPUT` and sent over the network to the remote process. Output from the remote command is copied back over the network and displayed on `SY$OUTPUT`.

Using RSH

Before you can successfully execute a remote command, the remote system must determine that you are allowed to do so. The RSH server checks the "R" services equivalence files to determine whether or not you are authorized to execute commands remotely. RSH normally uses the same authentication scheme as other "R" services. See the "R" Services Authentication and the "Host Equivalences" sections.

The following example shows how to use RSH to get a directory listing on the UNIX system UNIX.FLOWERS.COM from a local OpenVMS system:

```
$ RSH UNIX.FLOWERS.COM ls -l
```

This command assumes that the remote user name is the same as the local user name. To specify a

different remote user name, use the /USERNAME qualifier as shown in the following command:

```
$ RSH /USERNAME=zeno UNIX.FLOWERS.COM ls -l
```

If "R" services equivalence files are not set up, you can still use the RSH command by specifying the /PASSWORD qualifier. When a password is specified, rather than connecting to the RSH server, the RSH client connects to the REXEC server on the remote system. REXEC is identical in function to RSH, except that it uses a user name and password to perform authentication rather than equivalence files. The command format for specifying a password is as follows:

```
$ RSH /USERNAME=zeno /PASSWORD=race UNIX.FLOWERS.COM ls -l
```

Note! If you specify /PASSWORD without a value, you are prompted for the password.

You can modify where the remote command standard input is read and where standard output and standard errors are written. Normally, RSH uses SYS\$INPUT, SYS\$OUTPUT, and SYS\$ERROR for input, output, and error. You can redirect the input, output, or error streams using the /INPUT, /OUTPUT, or /ERROR qualifiers, respectively.

If you want to execute a command with RSH, but do not want your terminal to be tied up during the remote command execution, include the qualifier /INPUT=NLA0: on the RSH command to specify a null device. The remote command will see an end-of-file if it attempts to read from standard input.

Interrupting and Terminating RSH

Normally, RSH terminates when the remote command terminates. However, if you press **Ctrl/C** while RSH is running, the interrupt is sent to the remote process. If the remote command is being executed on a UNIX system, the **Ctrl/C** is perceived as an interrupt signal.

Logging Into a Remote System with RLOGIN

The RLOGIN command lets you interactively log into a remote system from your local system. RLOGIN is similar to TELNET, except that support for RLOGIN is not as widespread, and the authentication method relies on equivalence files that identify trusted hosts rather than passwords.

Using RLOGIN

If your user name is the same on the local and remote systems, or the "R" services equivalence files are set up appropriately, you can use the following command format to log in:

```
$ RLOGIN hostname
```

To use a different remote user name, use the following command format:

```
$ RLOGIN hostname /USERNAME=remote_user
```

Once an RLOGIN session has been established, the following character sequences typed at the beginning of a line have the effect described:

~.	A tilde followed by a period disconnects the session and exits RLOGIN.
~Ctrl/Z	A tilde followed by Ctrl/Z creates and connects you to a subprocess on the local system. When you log out of the subprocess, you return to your RLOGIN session.
~~	Two consecutive tildes transmit a single tilde to the remote system.

Terminating an RLOGIN Session

You terminate your session with the remote host by logging out as you normally would.

"R" Services Authentication

The "R" services RLOGIN, RSHELL, RCP, and RMT use *trusted users* and *trusted hosts* listed in two files on the destination system for access control: MULTINET:HOSTS.EQUIV and SYS\$LOGIN:.RHOSTS.

Host Equivalences

The MULTINET:HOSTS.EQUIV file (/etc/hosts.equiv on UNIX systems) provides a list of hosts to receive access on a system-wide basis. All users on the specified hosts can access the target system without specifying a user name or password. Each entry in this file consists of a host name.

Note! You cannot use the MULTINET:HOSTS.EQUIV file to allow access to an individual user; user names specified in this file are ignored.

The following example shows a sample HOSTS.EQUIV file.

```
localhost
sales.flowers.com
flowers.com
bubba.flowers.com
```

If the HOSTS.EQUIV file shown in the previous example exists on the system such as the example SALES.FLOWERS.COM, the following statements are true:

- Users on SALES.FLOWERS.COM will have RLOGIN, RCP, and RSHELL access to their own accounts on the system. (Allowed by the first two entries.)
- FLOWERS.COM and BUBBA.FLOWERS.COM are identified (in the last two entries) as trusted hosts, allowing any user on either of these systems to have RLOGIN, RCP, and RSHELL access to their own user name on SALES.FLOWERS.COM without specifying the user name or a password.

User Equivalences

The SYS\$LOGIN:.RHOSTS file (~/.rhosts on UNIX systems) allows remote users access to your user name. The format of an entry in this file consists of a host name and an optional user name:

hostname [*username*]

Each entry specifies that *username* on system *hostname* can access your user name on the target without specifying a password (you may omit *username* if your user names are identical on the two systems).

The following example contains an example .RHOSTS file.

```
flowers.com          system
unix.flowers.com     root
```

If the .RHOSTS file shown in the previous example belongs to the user FNORD on SALES.FLOWERS.COM, the following statements are true:

- The first entry grants access to user name FNORD on SALES.FLOWERS.COM from user SYSTEM on host FLOWERS.COM.
- The second entry grants access to user name FNORD from user ROOT on host UNIX.FLOWERS.COM.

Hence, either of these two remote users can use RLOGIN, RCP, or RSHELL to access FNORD's account on SALES.FLOWERS.COM without specifying a password.

Cautions Concerning Use of Equivalences

The following cautions apply when using "R" services equivalence files:

- When specifying a user in any authentication file (particularly on UNIX systems), make sure to specify the user name in the correct case. "ROOT" and "root" are treated as different user names on case-sensitive systems.
- The host initiating the RLOGIN, RCP, or RSHELL request must be listed in the destination host's host name database by DNS, or its name must be resolvable by DNS (if domain name service is enabled). If the destination host cannot determine the initiating host's name from the IP address in the connection request, it rejects the request.
- The resolved host name must be an exact match. For example, if the IP address resolves to FNORD.FOO.COM, it is not correct to put only FNORD in the HOST.EQUIV or .RHOSTS file. In addition to being fully qualified, entries must be of the same case.
- The MultiNet RLOGIN, RCP, and RSHELL servers cache the contents of the .RHOSTS and HOSTS.EQUIV files in memory for ten minutes to improve performance. This means changes to the .RHOSTS and HOSTS.EQUIV file may not be noticed by the network immediately. Your system manager can use the following command to flush the cache before the timeout period:

```
$ MULTINET NETCONTROL RLOGIN FLUSH
```

- Access control requirements differ between RLOGIN and other "R" services. RLOGIN requires both NETWORK *and* LOCAL access, while RSHELL, RMT, and RCP only require NETWORK access.

Logging Into a Remote System with TELNET

The MultiNet TELNET utility uses the standard Internet TELNET protocol to establish a virtual terminal connection between the interactive session on your OpenVMS system and a remote host. You can connect to any remote host on the network that supports the TELNET protocol, and perform any operation as if you were using a terminal physically connected to the remote host.

Refer to the *Accessing IBM Hosts with the TELNET Command* section for information on using the TELNET TN3270 and TN5250 features for accessing IBM hosts.

Starting a TELNET Connection

You can start TELNET and establish a connection to a remote host in either of two ways:

- From the DCL prompt
- Interactively from within the TELNET utility

The following example shows how to run TELNET and connect to a host in a single step.

```
$ telnet remote_host
Trying... Connected to remote_host, a host_type running os_type
```

In the next example, you invoke the TELNET utility. Once TELNET starts, you specify the remote host to which you want to connect.

```
$ telnet
ALTARF.PROCESS.COM MultiNet TELNET-32 4.3(103)
TELNET>connect remote_host
Trying... Connected to remote_host, a host_type running os_type
```

In either case, TELNET informs you of the CPU type and operating system software on the remote host (if that information is available from DNS or the host table).

Once you have logged in, proceed as though you were connected to the remote host via a locally attached terminal. Use the command syntax conventions native to the remote host.

Using TELNET Commands

You can only execute TELNET commands in command mode; that is, when you see the TELNET> prompt (before a connection is established) or the *host>* prompt (after a connection has been established).

You can force TELNET into command mode by entering the current escape character followed by an X. The default ESCAPE character is **Ctrl/^** (control-caret).

The following example shows how to force TELNET into command mode:

```
$ Ctrl/^ x
host>
```

Use the STATUS command to determine the state of all parameters associated with the TELNET session. The following example shows typical STATUS command output.

```
$ Ctrl/^ X
FLOWERS.COM>status

This is BUBBA.FLOWERS.COM, VAX/VMS Version V5.5
Connected to host IRIS.COM, a VAXSTATION-4000-60 running VMS via TCP.
Remote host is echoing
Host is not sending binary
Client is not sending binary
NO Abort Output character set
NO Interrupt Process character set
NO Are-You-There character set
NO Erase Character character set
NO Erase Line character set
Normal End Of Line mapping
Local Flow control
No log file
Remote host status reply:
KAOS::_VTA23: 11:24:21 (DCL) CPU=00.00.10.92 PF=322 IO=78 MEM=218
```

In general, when you type the TELNET ESCAPE character `Ctrl/^`, the next character you type is interpreted as follows:

?	Prints help information on TELNET escape commands.
A	Sends an "Attention" request to the remote host.
B	Sends a "Break" request to the remote host.
C	Closes the connection to the remote host.
O	Sends an "Abort Output" request to the remote host.
P	Spawns a new process (or attaches to a parent process, if there is one).
Q	Quits TELNET.
S	Prints the status of the TELNET connection.
T	Sends an "Are-You-There" request to the remote host.
X	Enters extended TELNET command mode.

To send the ESCAPE character itself to the remote host, type the ESCAPE character twice.

To change the ESCAPE character, use the DCL qualifier `/ESCAPE_CHARACTER`. For example, to change from the default ESCAPE character `Ctrl/^` to `Ctrl/A`, type:

```
TELNET>set escape "^A"

or:
```



```
$ TELNET /ESCAPE_CHARACTER="^A" flowers.com
```

You can determine all the available TELNET commands at any time by typing a question mark (?) at the TELNET> prompt.

Using TELNET Control Sequences

You can establish mappings between control characters and certain TELNET control sequences. This can often significantly improve terminal response. These mappings can also be used to provide a certain amount of system independence in the command interface across different systems. Consult the TELNET RFCs (854, 855, 856, 857, 1041, 1073, 1079, 1080, 1091) for additional information on TELNET control sequences (also known as IACs).

Normally, in a TELNET session, all characters typed at the terminal are inserted in the TELNET stream sequentially and interpreted sequentially at the remote system. Hence, even control characters that you want interpreted immediately (like **Ctrl/C** or **Ctrl/O** on an OpenVMS system) are interpreted on the remote system only after all characters that precede them in the command stream.

TELNET control sequences, however, can cause the remote system to perform their function before processing characters already in the input stream.

To specify control characters that map to these commands, specify them from the DCL command line:

```
$ TELNET /ABORT_OUTPUT="^O" flowers.com
```

or, using the SET command from within TELNET; for example:

```
TELNET>set abort-output "^O"
```

Table 5-1 summarizes the possible TELNET control sequences:

Table 5-1 TELNET Control Sequences

Sequence Name	Action	Equivalent OpenVMS Function
ABORT-OUTPUT	Cancels any output in progress and sends an Abort Output command to the TELNET server. Additionally, if the AUTO-FLUSH feature is enabled, a Timing Mark command is sent to the TELNET server; the TELNET client begins discarding any buffered output until a Timing Mark command is received in the response.	Ctrl/O
ARE-YOU-THERE	Sends an Are You There command to the TELNET server.	Ctrl/T
BREAK-CHARACTER	Sends a Break command to the TELNET server.	BREAK

Table 5-1 TELNET Control Sequences (Continued)

Sequence Name	Action	Equivalent OpenVMS Function
ERASE-CHARACTER	Sends an Erase Character command to the TELNET server.	<x
ERASE-LINE	Sends an Erase Line command to the TELNET server.	Ctrl/U
INTERRUPT-PROCESS	Sends an Interrupt Process command to the TELNET server.	Ctrl/C

You can also specify control characters from the DCL command line; for example:

```
$ telnet/abort_output=^O flowers.com
```

Running Applications over TELNET Connections

A TELNET connection normally exists between a remote pseudo-terminal (for example, NTYx:) and the TELNET user program. Characters received from the user's terminal are sent through the network to the remote pseudo-terminal and vice versa. Using the DCL qualifier /CREATE_NTY or the TELNET CREATE-NTY command, you can also connect the local end of the connection to a pseudo-terminal. Once the local end is connected to a pseudo-terminal, you can run other applications (such as KERMIT) over the TELNET connection.

The CREATE-NTY command first attempts to negotiate BINARY mode. BINARY mode ensures the connection is as transparent as possible. Then, a new NTYx terminal is created and the connection attached to it. Finally, the NTYx terminal is allocated to your current process and TELNET exits.

The following example shows how to use the DCL /CREATE_NTY qualifier.

```
$ TELNET/CREATE_NTY bubba
Trying... Connected to BUBBA, a VAX running VMS.
Welcome to BUBBA
Username: JOE
Password:
Welcome to VAX/VMS version V5.5 on node BUBBA
Last interactive login on Monday, 13-MAR-2000 13:34
Last non-interactive login on Tuesday, 14-MAR-2000 13:32
[ Process _VTA13: on BUBBA::VTA13: ]
$ Ctrl/^ X
BUBBA>create-nty
TELNET session now connected to _NTY3:
%DCL-I-ALLOC, _NTY3: allocated
$ kermit
VMS Kermit-32 version 3.3.111
```

```
Default terminal for transfers is: _TWA2:
Kermit-32>set line nty3:
Kermit-32>connect
[Connecting to _NTY3:. Type ^]C to return to VAX/VMS Kermit-32]
$
```

The following example shows how to use TELNET CREATE-NTY.

```
$ TELNET BUBBA
Trying... Connected to BUBBA, a VAX running VMS.
Welcome to BUBBA
Username: JOE
Password:
Welcome to VAX/VMS version V5.5 on node BUBBA
Last interactive login on Monday, 13-MAR-2000 13:34
Last non-interactive login on Tuesday, 14-MAR-2000 13:32
[ Process _VTA13: on BUBBA::VTA13: ]
$ Ctrl/^ X
BUBBA>CREATE-NTY
TELNET session now connected to _NTY3:
%DCL-I-ALLOC, _NTY3: allocated
$ kermit
VMS Kermit-32 version 3.3.111
Default terminal for transfers is: _TWA2:
Kermit-32>set line nty3:
Kermit-32>connect
[Connecting to _NTY3:. Type ^]C to return to VAX/VMS Kermit-32]
$
```

Accessing IBM Hosts with the TELNET Command

TELNET provides two IBM terminal emulations for accessing IBM hosts. The /TN3270 and /TN5250 qualifiers provide IBM 3270 and IBM 5250 terminal emulations, respectively. Using TELNET TN3270 and TN5250, you can:

- Log into IBM hosts
- Display and define your own keyboard map
- Capture screen output
- Print screen capture output

Both TN3270 and TN5250 modes use the OpenVMS screen management (SMG) runtime routines to create a full-screen IBM 3270 or 5250 mode display on your terminal. These TELNET modes give the appearance of being logged into the remote host from an IBM terminal.

Starting TELNET with an IBM Terminal Emulator

To start TELNET in TN3270 mode, enter the following command:

```
$ MULTINET TELNET /TN3270
```

To force TN3270 emulation, enter:

```
$ MULTINET TELNET /TN3270=FORCE
```

This qualifier is useful when communicating with a system that supports 3270 mode, but cannot negotiate it automatically, such as IBM mainframes running ACCESS/VMS. To start TELNET in TN5250 mode, enter:

```
$ MULTINET TELNET /TN5250
```

To force TN5250 emulation, enter:

```
$ MULTINET TELNET /TN5250=FORCE
```

Stopping an IBM Emulator Session

Exit a TN3270 or TN5250 session by pressing **Ctrl/C**.

IBM 3278 Models

In TN3270 mode, TELNET emulates an IBM 3278 terminal. The model number depends on the terminal "window" size (page width and length). The terminal (or window on a workstation) on which TN3270 mode TELNET is running must have at least 80 columns and 24 rows. Table 5-2 describes the actual emulation used, based on the terminal/window size.

Table 5-2 3278 Model Window Size

Minimum Size (Rows x Columns)	Emulated Terminal
24 x 80	3278 model 2
32 x 80	3278 model 3
43 x 80	3278 model 4
27 x 132	3278 model 5

TN5250 TELNET mode emulates a TN5251-11 terminal with 24 rows and 80 columns and has only one screen mode.

Mapping Your Keyboard

TN3270 and TN5250 modes use the OpenVMS SMG runtime routines and the files MULTINET:MAP3270.DAT and MULTINET:MAP5250.DAT, respectively, to perform terminal emulation on the local system. These files contain the terminal key sequence to IBM terminal key mappings for a wide variety of terminals. Only those terminals with entries in both MAP3270.DAT or MAP5250.DAT and the OpenVMS SMG terminal definition library (SYSS\$SYSTEM:TERMTABLE.TXT) can use the IBM terminal modes.

Displaying the Current Keyboard Mapping

Press the **HELP** key to display the current key mappings from the current key mapping data file (such as MAP3270.DAT). The help screen reformats and improves readability of the information

in the mapping file.

The following is an example help screen for MAP3270.DAT:

```
TN3270 Key Definitions (Press Help to dismiss)
PFK1  = "KP1" or "ESC 1"
PFK2  = "KP2" or "ESC 2"
PFK3  = "KP3" or "ESC 3"
PFK4  = "KP4" or "ESC 4"
PFK5  = "KP5" or "ESC 5"
PFK6  = "KP6" or "ESC 6"
PFK7  = "KP7" or "ESC 7"
PFK8  = "KP8" or "ESC 8"
PFK9  = "KP9" or "ESC 9"
PFK10 = "PF1 KP0" or "ESC 0"
PFK11 = "PF1 KP1" or "ESC -"
PFK12 = "PF1 KP2" or "ESC ="
PFK13 = "PF1 KP3" or "^F 1 3"
PFK14 = "PF1 KP4" or "^F 1 4"
PFK15 = "PF1 KP5" or "^F 1 5"
PFK16 = "PF1 KP6" or "^F 1 6"
PFK17 = "PF1 KP7" or "^F 1 7"
PFK18 = "PF1 KP8" or "^F 1 8"
PFK19 = "PF1 KP9" or "^F 1 9"
PFK20 = "PF2 KP0" or "^F 2 0"
PFK21 = "PF2 KP1" or "^F 2 1"
PFK22 = "PF2 KP2" or "^F 2 2"
PFK23 = "PF2 KP3" or "^F 2 3"
PFK24 = "PF2 KP4" or "^F 2 4"
PA1   = "ESC PF1" or "^P 1"
PA2   = "ESC PF2" or "^P 2"
LEFT  = "^H" or "LEFT"
RIGHT = "^L" or "RIGHT"
UP     = "^K" or "UP"
DOWN  = "^J" or "DOWN"
CLEAR = "^Z" or "KP_ENTER"
ENTER = "^M"
ESCAPE = "^C"
CAPTURE = "T" or "DO"
TAB    = "I"
BTAB   = "B"
INSRT  = " " or "ESC SPACE"
DELETE = "D"
ERASE  =
EEOF   = "E"
EINP   = "W"
HOME   = "KP_PERIOD"
```

The 3270.DAT file viewed without the HELP formatting is as follows:

```
vt100|vt200|vt220|vt240|vt200-80|vt300|vt400|vt100nam|pt100| {
enter   = '^m';
clear   = '^z'      | '\EOM'   | '\3M';
help    = '\E[28~'  | '\EH'    | '\C28~';
capture = '^t'      | '\E[29~' | '\C29~';
nl      = '^?';
tab     = '^i';
btabs   = '^b';
left    = '^h'      | '\E[D'   | '\EOD' | '\3D' | '\CD';
right   = '^l'      | '\E[C'   | '\EOC' | '\3C' | '\CC';
up      = '^k'      | '\E[A'   | '\EOA' | '\3A' | '\CA';
down    = '^j'      | '\E[B'   | '\EOB' | '\3B' | '\CB';
home    = '\EOn'    | '\3n';
fm      = '^y';
delete  = '^d';
eof     = '^e';
inp     = '^w';
insrt   = '^ '      | '\E ' ;

# pf keys
pfk1    = '\EOq'    | '\E1'    | '\3q';
```

```

pfk2  = '\EOr' | '\E2' | '\3r';
pfk3  = '\EOs' | '\E3' | '\3s';
pfk4  = '\EOt' | '\E4' | '\3t';
pfk5  = '\EOu' | '\E5' | '\3u';
pfk6  = '\EOv' | '\E6' | '\3v';
pfk7  = '\EOw' | '\E7' | '\3w';
pfk8  = '\EOx' | '\E8' | '\3x';
pfk9  = '\EOy' | '\E9' | '\3y';
pfk10 = '\EOP\EOp' | '\EO' | '\3P\3p';
pfk11 = '\EOP\EOq' | '\E-' | '\3P\3q';
pfk12 = '\EOP\EOr' | '\E=' | '\3P\3r';
pfk13 = '\EOP\EOs' | '^f13' | '\3P\3s';
pfk14 = '\EOP\EOt' | '^f14' | '\3P\3t';
pfk15 = '\EOP\EOu' | '^f15' | '\3P\3u';
pfk16 = '\EOP\EOv' | '^f16' | '\3P\3v';
pfk17 = '\EOP\EOw' | '^f17' | '\3P\3w';
pfk18 = '\EOP\EOx' | '^f18' | '\3P\3x';
pfk19 = '\EOP\EOy' | '^f19' | '\3P\3y';
pfk20 = '\EOQ\EOp' | '^f20' | '\3Q\3p';
pfk21 = '\EOQ\EOq' | '^f21' | '\3Q\3q';
pfk22 = '\EOQ\EOr' | '^f22' | '\3Q\3r';
pfk23 = '\EOQ\EOs' | '^f23' | '\3Q\3s';
pfk24 = '\EOQ\EOt' | '^f24' | '\3Q\3t';

# program attention keys
pal = '\E\EOP' | '^p1' | '\E\3P';
pa2 = '\E\EOQ' | '^p2' | '\E\3Q';

# local control keys

escape = '^c' | '^_'; # escape to telnet command mode
master_reset = '^g';

# local editing keys
settab = '\E';
deltab = '\E\';
clrtab = '\E:';
setmrg = '\E,';
sethom = '\E.';
coltab = '\E\E[B' | '\E\EOB' | '\E\3B' | '\E\CB';
colbak = '\E\E[A' | '\E\EOA' | '\E\3A' | '\E\CA';
indent = '\E\E[C' | '\E\EOC' | '\E\3C' | '\E\CC';
undent = '\E\E[D' | '\E\EOD' | '\E\3D' | '\E\CD';
} # end of vt100, etc.

```

On terminals without a **HELP** key, edit the **MAP3270.DAT** or **MAP5250.DAT** file and assign a value to the "help" function. For example, to assign the help function to either **Ctrl-X h** or **ESC h**, add this line to the file:

```
help = '^XH' | '\EH';
```

For VT-class terminals without a HELP key, TELNET supports **ESC h** by default. On these terminals, you do not need to modify the MAPxxx.DAT files.

Keyboard Mapping File Format

The keyboard mapping files contain mappings between characters entered from your keyboard, and 3270 or 5250 keycodes. The first line specifies all of the terminal types supported. For example, these mappings specify Compaq VT100-VT400 terminals:

```
vt100 | vt200 | vt200-80 | vt220 | vt240 | vt300 | vt400
```

Subsequent lines specify the IBM keycode followed by an equals sign (=) and the keystrokes (in single quotes) you press to send the keycode. Each key definition ends with a semicolon (;). Some reserved characters are:

- Caret (^) begins a **Ctrl** character sequence.
- Backslash and the letter "E" (\E) represents an ESCAPE character.
- Caret-question mark (^?) represents rub out.

For example, this key sequence:

```
delete = '^d';
```

sends the IBM DELETE code when you press **Ctrl/D**.

Functions

The following is a list of the TN3270 and TN5250 functions that can be used in the MAP3270.DAT and MAP5250.DAT files.

aplend	cursel	escape	left2	right	up
aploff	delete	ferase	lprt	right2	vertical_bar
aplon	deltab	fielndend	master_reset	sethom	werase
attn	disc	flinp	monocase	setmrg	wordbacktab
btabs	down	fm	nl	ettab	wordend
capture	dp	help	pal-pa3	space	wordtab
centsign	dvcnl	home	pcoff	synch	
clear	eeof	indent	pcon	tab	
clrtab	einp	init	pfk1-pfk36	test	
colbak	enter	insrt	reset	treq	
coltab	erase	left	reshow	undent	

Specifying Multiple Keystrokes

You can assign multiple keystrokes to a single code by separating each set of keystrokes with a vertical bar (|) operator. The following example sends the delete keycode to the host when you press either **Ctrl/D** or **Ctrl/?**.

```
delete = '^d' | '^?';
```

TN3270 Function Key Mapping

Table 5-3 lists the mappings between 3270 function keys and the keys on Compaq VT100, VT200, VT300, and VT400 series terminals.

Table 5-3 TN3270 Function Key Mappings

IBM Function	VT Terminal Key Sequences
Enter	Ctrl/M or RETURN
Clear	Ctrl/Z or ENTER
Input Editing Functions	
New line	DELETE
Tab	TAB or Ctrl/I
Backtab	Ctrl/B
Left	Ctrl/H or LEFT ARROW
Right	Ctrl/L or RIGHT ARROW
Up	Ctrl/K or UP ARROW
Down	Ctrl/J or DOWN ARROW
Home	Keypad
Delete	Ctrl/D
Erase to EOF	Ctrl/E
Erase Input	Ctrl/W
Insert	Ctrl/Space or ESC + Space
Attention Keys	
PA1	ESC + PF1 or Ctrl/P + 1
PA2	ESC + PF2 or Ctrl/P + 2
Local Control Keys	
TELNET Escape	Ctrl/C or Ctrl/[
Master Reset	Ctrl/G
Local Editing Keys	
Set Tab	ESC + ;

Table 5-3 TN3270 Function Key Mappings (Continued)

IBM Function	VT Terminal Key Sequences
Delete Tab	ESC + \
Clear Tabs	ESC + :
Set Merge	ESC + ,
Set Home	ESC + .
Column Tab	ESC + DOWN ARROW
Column Back Tab	ESC + UP ARROW
Indent	ESC + RIGHT ARROW
Unindent	ESC + LEFT ARROW
Function Keys	
PF1	Keypad 1 <i>or</i> ESC + 1
PF2	Keypad 2 <i>or</i> ESC + 2
PF3	Keypad 3 <i>or</i> ESC + 3
PF4	Keypad 4 <i>or</i> ESC + 4
PF5	Keypad 5 <i>or</i> ESC + 5
PF6	Keypad 6 <i>or</i> ESC + 6
PF7	Keypad 7 <i>or</i> ESC + 7
PF8	Keypad 8 <i>or</i> ESC + 8
PF9	Keypad 9 <i>or</i> ESC + 9
PF10	PF1 + Keypad 0 <i>or</i> ESC + 0
PF11	PF1 + Keypad 1 <i>or</i> ESC + -
PF12	PF1 + Keypad 2 <i>or</i> ESC + =
PF13	PF1 + Keypad 3 <i>or</i> Ctrl/F + 1 + 3
PF14	PF1 + Keypad 4 <i>or</i> Ctrl/F + 1 + 4
PF15	PF1 + Keypad 5 <i>or</i> Ctrl/F + 1 + 5
PF16	PF1 + Keypad 6 <i>or</i> Ctrl/F + 1 + 6

Table 5-3 TN3270 Function Key Mappings (Continued)

IBM Function	VT Terminal Key Sequences
PF17	PF1 + Keypad 7 <i>or</i> Ctrl/F + 1 + 7
PF18	PF1 + Keypad 8 <i>or</i> Ctrl/F + 1 + 8
PF19	PF1 + Keypad 9 <i>or</i> Ctrl/F + 1 + 9
PF20	PF2 + Keypad 0 <i>or</i> Ctrl/F + 2 + 0
PF21	PF2 + Keypad 1 <i>or</i> Ctrl/F + 2 + 1

Note! Key sequences denoted by **Keypad x** indicate key x on the VT terminal keypad.

TN5250 Function Key Mapping

Table 5-4 lists the mappings between 5250 function keys and the keys on Compaq VT100, VT200, VT300, and VT400 series terminals.

Table 5-4 TN5250 Function Key Mappings

IBM Function	VT Terminal Key Sequences
Enter	Ctrl/M <i>or</i> RETURN
Clear	Ctrl/Z <i>or</i> ENTER
Input Editing Functions	
New line	Del
Tab	Tab <i>or</i> Ctrl/1
Backtab	Ctrl/B
Left	Ctrl/H <i>or</i> Left arrow
Right	Ctrl/L <i>or</i> Right arrow
Up	Ctrl/K <i>or</i> Up arrow
Down	Ctrl/J <i>or</i> Down arrow
Home	Keypad .
Delete	Ctrl/D
Insert	Ctrl/Space <i>or</i> ESC + Space
Local Control Keys	

Table 5-4 TN5250 Function Key Mappings (Continued)

IBM Function	VT Terminal Key Sequences
TELNET Escape	Ctrl/C or Ctrl/[
Master Reset	Ctrl/G
Function Keys	
CMD1	Keypad 1 or ESC + 1
CMD2	Keypad 2 or ESC + 2
CMD3	Keypad 3 or ESC + 3
CMD4	Keypad 4 or ESC + 4
CMD5	Keypad 5 or ESC + 5
CMD6	Keypad 6 or ESC + 6
CMD7	Keypad 7 or ESC + 7
CMD8	Keypad 8 or ESC + 8
CMD9	Keypad 9 or ESC + 9
CMD10	PF1 + Keypad 0 or ESC + -
CMD11	PF1 + Keypad 1 or ESC + -
CMD12	PF1 + Keypad 2 or ESC + =
CMD13	PF1 + Keypad 3 or Ctrl/F + 1 + 3
CMD14	PF1 + Keypad 4 or Ctrl/F + 1 + 4
CMD15	PF1 + Keypad 5 or Ctrl/F + 1 + 5
CMD16	PF1 + Keypad 6 or Ctrl/F + 1 + 6
CMD17	PF1 + Keypad 7 or Ctrl/F + 1 + 7
CMD18	PF1 + Keypad 8 or Ctrl/F + 1 + 8
CMD19	PF1 + Keypad 9 or Ctrl/F + 1 + 9
CMD20	PF2 + Keypad 0 or Ctrl/F + 2 + 0
CMD21	PF2 + Keypad 1 or Ctrl/F + 2 + 1

Note! Key sequences denoted by **Keypad x** indicate key x on the VT terminal keypad.

Editing the Keyboard Mapping File

To customize a keyboard mapping file:

1	Copy the appropriate file (MAP3270.DAT or MAP5250.DAT) from the MULTINET: directory to your login directory; for example, USERS:[IGUANA]MAP3270.DAT.
2	Define the MAP3270 or MAP5250 logical name to point to that file instead of the version in the MULTINET: directory; for example: \$ DEFINE/JOB MAP3270 "@USERS:[IGUANA]MAP3270.DAT" Note! You must use the @ (at-sign) at the start of the file name.
3	Edit the file with any text editor. To test a particular entry for a terminal in the MAP3270 or MAP5250 file, define the KEYBD logical name for your entry; for example: \$ DEFINE KEYBD "my_new_vt420"

Capturing Screen Output and Printing Screen Captures

You can press the Do key at any time during a TN3270 or TN5250 session to store the contents of the current screen in a file in the current directory (the default directory when the TELNET session started). The output file is named TN3270.LIS or TN5250.LIS and captures only the current screen. Each time you press the Do key, a new version of this file is created.

For keyboards that do not have a Do key, assign a value to the capture function in the MAPxxxx.DAT file. For example, assign the capture function to accept Ctrl/T as follows:

capture = '^t'

On VT-style keyboards without a Do key, TELNET supports Ctrl/T by default. For these terminals, you don't need to modify the MAPxxxx.DAT files.

The MULTINET_TN3270_PRINTER logical name lets you direct TN3270 screen output to a print queue. To use this feature, enter:

\$ DEFINE MULTINET_TN3270_PRINTER queue_name

The MULTINET_TN5250_PRINTER logical name lets you direct TN5250 screen output to a print queue. To use this feature, enter:

\$ DEFINE MULTINET_TN5250_PRINTER queue_name

Using Transparent Mode

TN3270 supports a transparent mode similar to the transparent mode offered by the IBM 7171 ASCII device controller. This feature is enabled automatically by TELNET when transparent mode information is received from the IBM host. You can disable this feature before entering TN3270 with the following command:

```
$ DEFINE MULTINET_TN3270_TRANSPARENT_MODE DISABLED
```

Application Keypad Access for TN3270 and TN5250

You can enable or disable access to the application keypad in TN3270 mode with the MULTINET_TN3270_APPLICATION_KEYPAD logical name. The default value is ON. Disable access by defining the logical name as follows:

```
$ DEFINE MULTINET_TN3270_APPLICATION_KEYPAD OFF
```

You can enable or disable access to the application keypad in TN5250 mode with the MULTINET_TN5250_APPLICATION_KEYPAD logical. The default value is ON. Disable access by defining the logical as follows:

```
$ DEFINE MULTINET_TN5250_APPLICATION_KEYPAD OFF
```

TN3270 Emulation

The Yale Improved Null (/[NO]YALE) qualifier is enabled by default. Yale Improved Null replaces NULL characters found in fields with spaces when the TN3270 client writes the fields back to the server. Use the /NOYALE qualifier to disable this feature.

```
$ TELNET /TN3270/NOYALE
```

To disable text colors, use this command:

```
$ TELNET /TN3270/NOCOLOR
```

TN3270 Translation Table Mapping

TN3270 uses the MULTINET_TN3270_LANGUAGE logical to specify the regional language for the international character set translation table. Translation tables are stored in the TN3270.TRANSLATION file. When TELNET is invoked, the translation file is searched for in the SYS\$LOGIN directory. If it is not found, the MULTINET: directory is searched.

An entry in the translation table begins with the name of the language starting in the first column in the line. Use this value to define the MULTINET_TN3270_LANGUAGE logical. For example, this command specifies a translation table for a UK English keyboard:

```
$ DEFINE MULTINET_TN3270_LANGUAGE "UK_ENGLISH_DEC_MULTI"
```

The remainder of an entry consists of lines preceded with whitespace (either tabs or spaces). Each line contains these three values:

1	An EBCDIC+ code to be sent to the IBM host
2	The ASCII code to be displayed for that EBCDIC value
3	The ASCII character sent from the keyboard that causes the EBCDIC value to be sent to the host

A pound sign (#) specifies a comment and can appear in any column on a line, including lines containing translation codes. When specified on a line containing a translation code, the comment character must be preceded by at least one whitespace character. An entry is terminated by the first line following the entry that contains a "printable" character in column one. Entry names must start in the first column, and must consist only of uppercase letters, numbers, and the underbar sign. The maximum length of an entry name is 255 characters.

The file name of the translation table can be changed with the MULTINET_TN3270_TRANSLATION_TABLES logical. For example, to define a translation table named US_FOO.DAT, enter:

```
$ DEFINE MULTINET_TN3270_TRANSLATION_TABLES "US_FOO.DAT"
```

+ EBCDIC stands for Extended Binary-Coded-Decimal Interchange Code.

An error message is issued if either logical name, MULTINET_TN3270_LANGUAGE or MULTINET_TN3270_TRANSLATION_TABLES, points to a non-existent entry.

The following example contains a sample translation file. In this example, the first line of the UK_ENGLISH_DEC_MULTI entry indicates that for the EBCDIC character 0x5b, the ASCII character 0xa3 is displayed. When the ASCII character 0xa3 is received from the keyboard, the EBCDIC character 0x5b is sent to the host.

```
#
# UK EBCDIC mapped into The DEC Multinational Character Set
# Use following command to specify this table:
# $ DEFINE MULTINET_TN3270_LANGUAGE "UK_ENGLISH_DEC_MULTI"
#
UK_ENGLISH_DEC_MULTI
0x5b 0xa3 0xa3 # British monetary pound sign
0x4a 0x24 0x24 # Dollar sign ($)
#
#
# Austrian German mapped into The DEC Multinational Character
# Set. Use following command to specify this table:
# $ DEFINE MULTINET_TN3270_LANGUAGE "AUSTRIAN_GERMAN_DEC_MULTI"
#
#
AUSTRIAN_GERMAN_DEC_MULTI
0x4a 0xc4 0xc4 # A with umlaut
0x5a 0xdc 0xdc # U with umlaut
0x6a 0xf6 0xf6 # o with umlaut
0x79 0x60 0x60 # Grave
0x5b 0x24 0x24 # Dollar sign
0x7b 0x23 0x23 # Hash sign
0x7c 0xa7 0xa7 # Section sign
0x5f 0x5e 0x5e # Carat sign
0xa1 0xdf 0xdf # Beta sign
0xc0 0xe4 0xe4 # a with umlaut
0xd0 0xfc 0xfc # u with umlaut
0xe0 0xd6 0xd6 # O with umlaut
```

```
0x4f 0x21 0x21 # Exclamation point
0x7f 0x22 0x22 # Double quote
```

Troubleshooting TELNET

This section describes common problems that can occur when using TELNET to connect to a remote host.

Connection Problems

If you cannot connect to the remote host, use PING as follows to discover any network problems. For information about starting PING, refer to the *Administrator's Reference*.

1	Ping the loopback address of your workstation, 127.0.0.1 to verify that MultiNet is working properly and that it can send and receive messages.
2	Ping your workstation by its IP address to verify that it is recognized on the network.
3	Ping your workstation by its host name to verify that it is recognized on the network and that its host name is being resolved.
4	Ping the broadcast address on your network to verify that your network can broadcast messages.
5	Ping another host on the same network by IP address to verify that the workstation can communicate with other hosts on the network.
6	Ping another host on the same network by host name to verify that host names are being resolved.
7	Ping a host on a different network, first by IP address and then by host name, to verify the default route is correct and that host names are being resolved.

Problems Logging In

If you cannot log into the remote host:

1	Make sure you have a valid user name on the remote host.
2	Make sure you are entering the correct user name and password. If you still have difficulties logging in, contact your network administrator.

Chapter 6

Remote File Access with the RCP, FTP, and TFTP Utilities

This chapter describes how to copy files between your local system and a remote system using the RCP, FTP, and TFTP utilities, and covers the following topics:

- Using the RCP utility to only copy files
- Using the FTP utility to copy files between the local and remote hosts
- Using the TFTP utility to only copy files

The FTP commands for renaming files, deleting files, and creating and deleting directories are described in the FTP command reference in Appendix B.

Copying Files Using RCP

The MultiNet RCP utility uses the 4.3BSD UNIX "rcp" (remote copy) protocol to transfer files between the local host and a remote host. The Kerberos version of RCP also provides authenticated access between the two systems.

When the index file creates new buckets (the space allocated to store units of data) beyond the previous End-Of-File mark, but the End-Of-File is not updated to reflect the new buckets, RCP transfers the allocated buckets to the End-Of-File. You can turn this feature off by defining the logical `MULTINET_RCP_INDEX_UPTO_EOF`.

Requirements for RCP

The requirements for using the RCP utility are:

- Both the local and remote host must support the rcp protocol.
- You must specify the names of files on the remote host using the file-naming conventions of the remote host.
- If the remote host is an OpenVMS system, you must ensure that neither the system-wide login command procedure nor your local LOGIN.COM file displays any text. See Section Inhibiting Output from SYLOGIN.COM and LOGIN.COM for more information on inhibiting output from

these command procedures.

The "R" services authentication database files on the server system must be configured to allow RCP access from the local system. See the *Using RCP* section for additional information on "R" services authentication.

Using RCP

You can use RCP interactively or via a command file in batch mode.

Before you can copy files using RCP, the remote system must determine that you are allowed to do so. Normally, the remote system's RCP server checks the "R" services host equivalence files to determine whether or not you are authorized to copy files to or from the remote system. RCP uses the same authentication scheme as RLOGIN and RSHELL. (See Chapter 5 for information about RCP authentication and the host equivalence files.)

However, if you are using RCP with Kerberos authentication, authentication is handled by acquiring "tickets" that permit access to cooperating systems. (See Chapter 4 for more information.)

The following is an example using RCP to copy the file /etc/hosts from the UNIX system UNIX.SPROCKETS.COM to the user's current default directory on the local OpenVMS system.

Note! The double quotation marks around "/etc/hosts" are necessary to prevent the slashes in the path name from being interpreted by DCL.

```
$ RCP UNIX.SPROCKETS.COM::"/etc/hosts" [ ]
```

This command assumes the remote user name is the same as the local user name. To specify a different remote user name, use the /USERNAME qualifier as shown in the following command:

```
$ RCP /USERNAME=JETSON UNIX.SPROCKETS.COM::.cshrc [ .UNIX-FILES ]
```

If the host equivalence files are not set up, you can still use the RCP command by specifying the /PASSWORD qualifier. In that case, REXEC authentication is used instead. The command format for specifying a password is as follows:

```
$ RCP /USERNAME=JETSON /PASSWORD=ASTRO -  
_ $ UNIX.SPROCKETS.COM::report.july [ .REPORTS ]
```

Note! If you specify /PASSWORD without a value, you are prompted for the password with echoing disabled.

To copy files with RCP using Kerberos authentication, use the following format:

```
$ RCP /AUTHENTICATION=KERBEROS UNIX.SPROCKETS.COM::"etc/hosts" [ ]
```

or

```
$ RCP /AUTHENTICATION UNIX.SPROCKETS.COM::"etc/hosts" [ ]
```

Inhibiting Output from SYLOGIN.COM and LOGIN.COM

The rcp protocol requires that neither the system-wide login command procedure (SY\$MANAGER:SYLOGIN.COM) nor users' LOGIN.COM procedures display any output. The following example shows commands to add to your LOGIN.COM and the system-wide SYLOGIN.COM to prevent any output from being displayed when they are executed.

```
$ VERIFY = 'F$VERIFY(0)                ! Turn off verify without echoing
$ IF F$MODE() .EQS. "OTHER" THEN EXIT  ! If a DETACHED process (RSHELL)
.
.
.
$ IF VERIFY THEN SET VERIFY             ! If a batch job, may want to turn
                                         ! verify back on.
```

Accessing Files with FTP

The FTP utility uses the Internet standard File Transfer Protocol (FTP) to transfer files between the local host and a remote host. FTP also allows you to perform directory and file operations, such as changing the working directory, listing files, renaming directories and files, and deleting directories and files.

The FTP utility has a command-line interface. Each action, such as copying files, requires a specific command.

Requirements for Using FTP

Requirements for using the FTP utility include the following:

- Both the local and remote host must support the Internet standard File Transfer Protocol.
- The names of files on the remote host must be specified using the file-naming conventions of the remote host.

Invoking FTP and Logging In

You can use FTP interactively or in batch mode with a command file.

When you invoke FTP, an FTP server process is created on the remote host. You can perform a limited set of operations on the files and directories that you have permission to access. FTP authenticates you on the remote host by checking the user name and password you specify against those in the authorization database on the remote host. For simplicity in this discussion, this verification process is referred to as *logging in*; however, you do not actually log in interactively to the remote host.

To illustrate, assume you are a user on the local system and you want to log into the remote host RESEARCH.FLOWERS.COM. You can log in as yourself (by entering your name) or you can log in as any other user on RESEARCH, for example, "MARK" or "BUBBA," as long as the specified user name is valid on the remote host and you know Mark's or Bubba's password.

Note! Even though logging into another user's account is mentioned in the previous section, sharing

passwords with other users is strongly discouraged.

You can connect to RESEARCH either by specifying the host name at the DCL command prompt (see Example 6-1), or by entering the CONNECT command at the FTP prompt (see Example 6-2).

Example 6-1 Specifying Host Name at DCL Prompt

```
$ FTP RESEARCH.FLOWERS.COM
DEVELOPMENT.FLOWERS.COM MultiNet FTP user process 4.3(nnn)
Connection opened (Assuming 8-bit connections)
<RESEARCH.FLOWERS.COM MultiNet FTP Server Process 4.3(nnn) at
Mon 13-Mar-2000 7:42am-EST
RESEARCH.FLOWERS.COM>LOGIN MARK
Password: password [not displayed]
RESEARCH.FLOWERS.COM>
```

Example 6-2 Enter Connect Command at FTP Prompt

```
$ FTP
DEVELOPMENT.FLOWERS.COM MultiNet FTP user process 4.3(nnn)
FTP>CONNECT RESEARCH.FLOWERS.COM
Connection opened (Assuming 8-bit connections)
<RESEARCH.FLOWERS.COM MultiNet FTP Server Process 4.3(nnn) at
Mon 13-Mar-2000 7:42am-EST
RESEARCH.FLOWERS.COM>LOGIN MARK
Password: password [not displayed]
RESEARCH.FLOWERS.COM>
```

Note! The initial FTP prompt (before connection to the remote host) is FTP>. After a connection is established, the prompt changes to the name of the remote host and FTP enters command mode.

At this point, you can specify your user name and password on RESEARCH with the FTP LOGIN command. Alternately, you can enter a command such as "LOGIN MARK" to log in as Mark (assuming you know Mark's password). The system then displays the "Password:" prompt. After you enter the password (which is not echoed), the system returns to FTP command mode, displays the prompt, and awaits further input.

Each time you invoke FTP, it checks first for a file called FTP.INIT in your login directory (SYS\$LOGIN) and executes any commands in that file before it prompts you for input. Any commands you want executed at the beginning of every FTP execution can be included in this file. See the *FTP Initialization File* section for a description of FTP commands commonly used in FTP.INIT files.

Note! Because the FTP server process is started by running SYS\$SYSTEM:LOGINOUT.EXE, both the system-wide login command procedure (SYS\$MANAGER:SYLOGIN.COM) and the specific

user's LOGIN.COM are executed. As a result, any customization such as specifying default file protection, or process/job logical name definitions, and so on, are invoked in these command procedures and are available under the FTP server process.

All standard OpenVMS security-checking mechanisms are used to validate the FTP server process creation. If either of these command procedures contain any commands that are specific to interactive jobs (SET TERMINAL commands, for example), the FTP server process may crash. The easiest way to avoid this problem, without altering the functionality of these command procedures, is to use the DCL lexical function F\$MODE together with interactive specific commands. For example:

```
$ IF F$MODE( ) .EQS. "INTERACTIVE" THEN SET TERMINAL /INQUIRE
```

The *FTP Log Files* section provides more information to assist you in determining the cause of any problems with the FTP server.

Using FTP Commands

After you have logged into a remote host, as described in the *Invoking FTP and Loggin In* section, you can use FTP commands for operations such as copying files between hosts, changing working directories, listing directories, removing files, and renaming files. All FTP commands are described in Appendix B.

The FTP user interface looks very similar to the Compaq Computer TOPS-20 command interface. In particular:

- You can type an **ESC** (escape character) at any point to attempt to complete (fill in) the current command, parameter (including file names), or qualifier.
- You can type a question mark (?) at any time for help on what to enter next.
- A question mark entered at the current FTP prompt displays the currently available commands. The commands that are available depend on whether or not a connection to a remote server has been established. Some commands are always recognized; others are recognized only before or after a connection has been made.

Getting FTP Command Help

The HELP command displays a brief description of a specified FTP command, general help information, or a list of available HELP topics. The format of the HELP command is as follows:

```
FTP>HELP [command]
```

If you specify the command name, HELP displays information for the specified command. If you type a ? in place of a command, HELP displays general help information. If you request HELP without an argument, the HELP facility lists available help topics and instructions for obtaining additional information.

Note! The available commands vary depending on whether you have an open connection to a remote host.

Using Basic FTP Commands

Some commands simply set or reset various FTP options. They can be explicitly set using the ON argument or reset using the OFF argument. The default, if no argument is typed, is TOGGLE. Hence, if an option is on, executing the command controlling the option sets it to off. Executing the command a second time resets it to on. For example, when you first invoke FTP, the VERBOSE option (which gives detailed messages) is off. The following command would toggle VERBOSE on:

```
FTP>VERBOSE
```

You can reset the VERBOSE option to off by executing the above command a second time, hence "toggling" the setting back and forth.

You can display the state of a MultiNet FTP Server at any given time using the STATUS command. The following example shows the information reported by the STATUS command. Note, however, that some FTP implementations do not support the STATUS command.

```
RESEARCH.FLOWERS.COM>STATUS
<RESEARCH.FLOWERS.COM MultiNet FTP Server Process 4.3(nnn)
User MARK logged into directory USERS:[MARK]
<The current transfer parameters are:
<  MODE S
<  STRU O VMS
<  TYPE A N
<A connection is open to host DEVELOPMENT.FLOWERS.COM
<The data connection is CLOSED.
```

Specifying TCP Window Size with FTP

The FTP Server and Client let you specify the TCP window sizes to use during an FTP transfer. The value to be used is determined as follows:

Table 6-3 TCP Window Size During an FTP Transfer

If...	Then use...
The logical name MULTINET_FTP_WINDOW_SIZE is defined	Its equivalence string as the value.
The /WINDOW_SIZE qualifier is specified with FTP [/SERVER]	The value specified with the qualifier.
A value is specified with [SITE] WINDOW-SIZE size	The value specified.

If none of these criteria exist, then use the default value 32768.

In all cases, the value must be between NET_MIN_TCPWINDOW and NET_MAX_TCPWINDOW (presently 512 and 1073741824, respectively). The size of the send and receive buffers is set to the specified value.

File Name Translations

When you issue an FTP GET command to a host running the UNIX Operating System and you do not specify an output file name, the resulting VMS file name can contain unexpected characters. These characters occur because the UNIX Operating System has case-sensitive characters and special symbols that require conversion before they can be used with VMS.

You can use the /FDL qualifier with the FTP client GET and PUT commands for compatibility with DEC TCP/IP Services for OpenVMS (formerly UCX). When you create a file with the PUT /FDL qualifier, a file description language (FDL) file is created at the same time as the original file. The contents of the original file are transmitted in IMAGE (binary) mode.

The FDL file has the same name except that "FDL" is appended to the file name extension.

An example of the PUT command is:

```
host>PUT /FDL AFILE.TXT BFILE.TXT
<ASCII Store of USERS:[ME]BFILE.TXTFDL;1 started.
<Transfer completed. 888 (8) bytes transferred.
<IMAGE Store of USERS:[ME]BFILE.TXT;1 started.
<Transfer completed. 6 (8) bytes transferred.
```

This command copies AFILE.TXT to BFILE.TXT on the system to which you are connected, then creates another file, BFILE.TXTFDL.

The BFILE.TXTFDL file is in ASCII format and resembles:

```
IDENT    " 13-MAR-2000 17:13:24    VAX/VMS FDL$GENERATE Routine"
SYSTEM

FILE      SOURCE                     VAX/VMS
          ALLOCATION                  5
          BEST_TRY_CONTIGUOUS       no
          BUCKET_SIZE                0
          CONTIGUOUS                 no
          DEFERRED_WRITE             no
          EXTENSION                   0
          GLOBAL_BUFFER_COUNT        0
          MT_BLOCK_SIZE              512
          MT_PROTECTION               32
          MAX_RECORD_NUMBER           0
          MAXIMIZE_VERSION           no
          NAME                        "USERS:[ME]AFILE.TXT;1"
          ORGANIZATION               sequential
          OWNER                       [STAFF,ME]
          PROTECTION                  (system:RWED, owner:RWED,group:,world:)
          READ_CHECK                  no
          SUPERSEDE                   no
          WRITE_CHECK                 no
RECORD    BLOCK_SPAN                 yes
          CARRIAGE_CONTROL            carriage_return
          CONTROL_FIELD_SIZE          0
```

FORMAT	variable
SIZE	0

The newly created BFILE.TXT file is in raw block format which is not easily readable. When you use the GET /FDL command to retrieve the file, the original format is restored using the attributes stored in the FDL file. If you do not use the /FDL qualifier with the GET command, the new raw block format is retained.

In all instances, the FDL file is retained and must be deleted independently.

Notes:

- The FTP server /TYPE=EBCDIC qualifier is no longer supported.
- If you invoke FTP from the DCL command line and a password string is case-sensitive, use the following format for the command:

```
$ FTP /USER=username /PASSWORD=" "MiXedCAsE" " "
```

If you don't use quotation marks, MultiNet converts the password to lowercase.

- If you replaced the FTP_SERVER.COM file, you must add /ACCESS=NOSPAWN on "captive" accounts such as the ANONYMOUS account so that users cannot spawn commands. Spawning commands from such accounts opens a potential security hole.
- When transferring files between OpenVMS systems, do not use the BINARY command except when the desired output requires fixed, 512-byte records; most importantly, do not use BINARY on Process Software ECO save sets that you acquired with FTP, if you are using FTP from a MultiNet system.

The following table shows how UNIX Operating System printable file name characters are translated into VMS file names:

VMS Character Value	Server Char.	Hex	VMS Character Value	Server Char.	Hex	VMS Character	Server Char.	Hex Value
\$4A	^A	1	\$5A	!	21	\$7A	Space	20
\$4B	^B	2	\$5B	“	22	\$7B	;	3B
\$4C	^C	3	\$5C	#	23	\$7C	<	3C
\$4D	^D	4	\$5E	%	25	\$7D	=	3D
\$4E	^E	5	\$5F	&	26	\$7E	>	3E
\$4F	^F	6	\$5G	‘	27	\$7F	?	3F
\$4G	^G	7	\$5H	(28			
\$4H	^H	8	\$5I)	29	\$8A	@	40

VMS Character Value	Server Char.	Hex	VMS Character Value	Server Char.	Hex	VMS Character Value (Continued)	Server Char.	Hex Value
\$4I	^I	9	\$5J	*	2A	\$8B	[5B
\$4J	^J	A	\$5K	+	2B	\$8C	\	5C
\$4K	^K	B	\$5L	,	2C	\$8D]	5D
\$4L	^L	C	\$5N	.	2E	\$8E	^	5E
\$4M	^M	D	\$5O	/	2F			
\$4N	^N	E	\$5Z	:	3A	\$9A	'	60
\$4O	^O	F	\$			\$9B	{	7B
\$4P	^P	10	\$6A	^@	00	\$9C		7C
\$4Q	^Q	11	\$6B	^[1B	\$9D	}	7D
\$4R	^R	12	\$6C	^\	1C	\$9E	~	7E
\$4S	^S	13	\$6D	^]	1D	\$9F	DEL	7F
\$4T	^T	14	\$6E	^^	1E			
\$4U	^U	15	\$6F	^-	1F			
\$4V	^V	16						
\$4W	^W	17						
\$4X	^X	18						
\$4Y	^Y	19						
\$4Z	^Z	1A						

- International characters in the range of octal 200 to 377 are translated as a dollar sign (\$) followed by the three-digit octal value for the character.
- Directory names copied to VMS are appended with the ".DIR" suffix.
- The dot (.) character is treated as a special case. The first occurrence in a file name is interpreted explicitly as a dot; the next occurrences are translated into the "\$5N" character sequence shown in the previous table. In a directory name, all occurrences of the dot character are translated into the "\$5N" character sequence.
- A dollar sign followed by a letter indicates that the case should be shifted from its current state.

An example of file name translation occurs when a UNIX file called "foo.bar#1.old" is copied to the VMS system. The resulting VMS file name is "FOO.BAR\$5C1\$5NOLD". If the file was a

directory, the translated name would be "FOO\$5NBAR\$5C1\$5NOLD.DIR". If the UNIX file name was "Foo.BAr#1.old", the translated case-sensitive VMS file name would be "\$F\$OO.\$BA\$R\$5C1\$5NOLD".

Listing the Contents of a File

You can use the GET command to list the contents of a file as follows:

```
$ GET filename TT:
```

This command displays a list of the files on your terminal, and works with all FTP servers.

Working with Directories

When you open a connection to a remote host and log in, your default directory is set to your login directory on the remote system. If you log in as another user, your default directory is set to that user's login directory. You can find out the path name of this directory with the command:

```
FTP>PWD
```

You can list the contents of your current working directory on the remote host with the command:

```
FTP>DIR
```

You can change the working directory on the remote host to remote_directory with the command:

```
FTP>CD remote_directory
```

To change the working directory on the local host to local_directory, use the command:

```
FTP>LCD local_directory
```

Commands for Copying Files

The GET and PUT commands are the two basic commands for copying files between your system and a remote host. The GET command copies a single file from the remote host to your system. The PUT command copies a single file from your system to the remote host. These commands have the following format:

```
FTP>GET remote_file local_file
```

```
FTP>PUT local_file remote_file
```

Under OpenVMS, the GET and PUT commands create new files. For other operating systems, the file is only created if it does not exist; if the file exists, an error displays. The AGET and APUT commands can be used to append to an existing file. These two commands have the following format:

```
FTP>AGET remote_file local_file
```

```
FTP>APUT local_file remote_file
```

The GET and PUT commands copy single files. Their counterparts, MGET and MPUT, copy

multiple files. The format of these commands is similar, but not identical, to that of GET and PUT:

```
FTP>MGET remote_file
FTP>MPUT local_file
```

In these two commands, you specify the file names with wildcard specifications. For MGET, use the file name wildcard syntax for the remote host. For MPUT, use the OpenVMS file name wildcard syntax. The files retain their original names when they are copied. An MGET to an empty directory returns a status code of 552 from the FTP server.

Parameters for Copying Files

Transfer parameters define how a file should be copied. The three transfer parameters and their values are described in the following list:

STRUCTURE

Defines the structure of files to be transferred; takes one of the following values:

FILE	An unstructured byte stream. This is the default when communicating with systems that do not understand the OpenVMS structure described in the <i>FTP VMS Structure</i> section.
RECORD	A file that is partitioned into records.
VMS	An arbitrary OpenVMS file; allows for transparent transfer of any RMS file between cooperating systems.

Note! The "VMS" transfer structure is automatically negotiated between systems that support it. After connecting to a remote system, the MultiNet FTP utility sends the FTP command "STRU O VMS" to the FTP server. If the server responds positively, both sides use the "VMS" structure to ensure total transparency when transferring files (that is, all RMS record and file attributes are retained). If the server responds negatively, both sides default to the "FILE" transfer structure.

FTP VMS Structure

TYPE

Defines the contents of files to be transferred; takes one of the following values:

ASCII	A file consisting of ASCII characters (the default).
BACKUP	Like IMAGE, but causes the local file to be written with 2048-byte fixed length records; used for transferring OpenVMS BACKUP savesets.
IMAGE	A binary image.
LOGICAL-BYTE	Used for doing binary transfers with TOPS-20 systems.

MODE

Defines how the file should be transferred; takes one of the following values:

COMPRESSED	Run length-encoded compression.
STREAM	Normal data transfer (the default).

FTP commands copy files using the current transfer parameters. When you first start FTP, the default transfer parameters are **FILE** structure, **ASCII** type, and **STREAM** mode. **VMS** structure is used if the FTP Server supports it. Use the following commands to change the transfer parameters from their defaults:

```
FTP>TYPE type_name
FTP>STRUCTURE struct_name
FTP>MODE mode_name
```

There are a number of command synonyms for the **TYPE** and **STRUCTURE** commands; see Appendix B for a complete list.

FTP Commands While a Transfer is in Progress

Control characters entered during an FTP file transfer have the following effects:

Press...	To
Ctrl/G	Send an abort command to the remote server, thus aborting a data transfer.
Ctrl/A	Display the state and progress of the file transfer.
Ctrl/P	Suspend the transfer and spawn a new DCL subprocess. The file transfer will continue upon return to the FTP program from the spawned DCL subprocess.

Aborting a file transfer does not work correctly with servers that do not support the **ABOR** (abort) command. If attempted, the connection to the server may be lost.

Issuing FTP Commands From the DCL Command Line

You usually run the FTP utility by typing the FTP command then issuing additional commands once the program starts. If you are only interested in transferring one file, or issuing a single FTP command, you can specify the command on the DCL command line. See **MULTINET FTP** in Appendix A for the complete DCL command syntax.

For example, if you wish to retrieve the file "pub/hack.c" via anonymous login to the host **FLOWERS.COM**, you might issue the DCL command:

```
$ FTP /USER=ANONYMOUS /PASSWORD=GUEST FLOWERS.COM GET pub/hack.c hack.c
```

To get a listing of the "pub" directory on this same system, you would use the command:

```
$ FTP /USER=ANONYMOUS /PASSWORD=GUEST FLOWERS.COM DIR pub
```

If you want to retrieve all files in the "pub" directory and copy them to your current directory on your local system, you might use the command:

```
$ FTP /USER=ANONYMOUS /PASSWORD=GUEST FLOWERS.COM MGET pub/*
```

FTP Command Scripts

FTP commands are usually entered directly from the keyboard. You can, however, execute a predefined sequence of FTP commands by redirecting standard input (SYS\$INPUT) interactively, or from within a DCL command procedure.

The following example shows an interactive session that uses a predefined command script, in this case in the file FTP.COM, to control FTP:

```
$ FTP /TAKE=FTP.COM
```

The following example shows a sample FTP.COM file. The italicized comments are provided only to explain each line in the FTP.COM file; do not include them in the actual file!

```
SET FLOWERS.COM /USER:BOOJUM /PASS:SNARK      Set user & password
CONNECT FLOWERS.COM                          Open connection
GET FOO.BAR NEWFOO.BAR                       Execute an FTP command
EXIT                                          Conclude session
```

The following example shows a DCL command procedure that runs FTP to get the file FOO.BAR from the remote host FLOWERS.COM.

```
$! FTP DCL command procedure
$ FTP
SET FLOWERS.COM /USER:BOOJUM /PASS:SNARK
CONNECT FLOWERS.COM
GET FOO.BAR NEWFOO.BAR
EXIT
$! continue with any other commands
```

Ending an FTP Session

Once you have finished with your FTP session, you can either break the connection with the remote system while still remaining in FTP command mode, or you can log out from the remote host, exit FTP, and return to DCL.

To close the current connection without terminating in FTP, enter the command:

```
FTP>BYE
FTP>
```

To close the connection and return to DCL, enter the command:

```
FTP>EXIT
$
```

FTP Log Files

The MultiNet FTP Server keeps a log of all FTP transactions that occur between the client and server after login in the file FTP_SERVER.LOG in the login directory on the server system. The following sample log file contains the FTP transactions involved in a user logging in under the user name SMITH, issuing a "DIRECTORY" command, and then retrieving the file "FOO.BAR."

Note! If the MultiNet FTP server process does not start or mysteriously disappears, examine the beginning of the FTP_SERVER.LOG file for any error messages.

Because the system-wide login command procedure (SYS\$MANAGER:SYLOGIN.COM) and the user's LOGIN.COM are executed as part of the server process creation, any errors in these procedures can cause the server process to die suddenly. In most instances, however, the reason for the process terminating will appear at the beginning of the FTP_SERVER.LOG file.

```
-----
FTP Login request received at Mon Mar 13 15:30:27 2000
      from remote IP address 127.0.0.1
-----
>>> 230 User SMITH logged into U1:[SMITH] at Mon 13-Mar-00 15:30, job 3a.
<<< TYPE A
>>> 200 Type A ok.
<<< STRU F
>>> 200 Stru F ok.
<<< MODE S
>>> 200 Mode S ok.
<<< PORT 127,0,0,1,4,14
>>> 200 Port 4.14 at Host 127.0.0.1 accepted.
<<< LIST
>>> 150 List started.
>>> 226 Transfer completed.
<<< PORT 127,0,0,1,4,15
>>> 200 Port 4.15 at Host 127.0.0.1 accepted.
<<< RETR foo.bar
>>> 150 ASCII retrieve of USERS:[SMITH]FOO.BAR;1 started (210 bytes).
>>> 226 Transfer completed. 210 (8) bytes transferred.
<<< QUIT
>>> 221 QUIT command received. Goodbye.
SMITH job terminated at 13-MAR-2000 15:31:23.08
```

Anonymous FTP

Many system managers use "anonymous FTP" to allow network access to files of general interest on their system, without having to assign a user name to each user who wants access to the files. Anonymous FTP means that the ANONYMOUS login is created on a system to permit anyone access to that system. When using anonymous FTP, connect to the remote system as you normally

would, but instead of specifying your user name, specify the user name "anonymous" and the password "guest." In many implementations, you are restricted to read-only access of the files in a certain directory or a certain directory tree.

Note! While many systems allow you to use any password, some systems only allow anonymous FTP access with the password "guest." Many systems prefer you to enter your e-mail address (username@host) instead of the "guest" password; either method works. Also, specify the "anonymous" user name in lowercase, as many systems (primarily those running UNIX) support case-sensitive user names. Hence, "anonymous" and "ANONYMOUS" are considered different user names, and only the former can be used for anonymous FTP access.

Transferring Files From Behind a Firewall

The MultiNet FTP Client **PASSIVE** command allows a range of control of the **PASV** directive for transferring files from FTP servers when your system is located behind a "firewall" gateway. The list of parameters and an explanation of how they work follows:

- an **ON** parameter (the default setting)
- an **OFF** parameter
- a **NEGOTIATED** parameter (the default setting)
- a **/PASV DCL** qualifier, allows you to specify the **PASSIVE** command setting as you start up the FTP Client (at the **FTP>** prompt, you may specify either **PASSIVE** or **PASV**; the two are interchangeable)

Note! If the change in the default setting causes you problems or changes the way things have worked for you in the past, you may control the default setting for your site by putting the appropriate **PASSIVE** command in the file **MULTINET:FTP.INIT**.

With **PASSIVE** mode **ON**, the Client sends the **PASV** directive to the Server, instructing it to wait for the Client to make the data connection. If the Server does not understand the **PASV** command, the connection is aborted. The default for **PASSIVE** is **ON** to help facilitate transfers through a firewall. Under certain conditions, this default might cause problems. Use the new MultiNet FTP client logical **MULTINET_FTP_NONPASV** to turn off the **PASSIVE** mode default or use the passive command on the command line. When you define this logical, passive mode is not used as the default.

With **PASSIVE** mode **OFF**, the FTP Client expects the FTP Server to establish the connection over which data is transferred. (Note that this may not work through firewalls as some FTP Servers do not support the **PASSIVE** command.)

With **PASSIVE** mode **NEGOTIATED**, the FTP Client sends the **PASV** command as with **PASSIVE** mode **ON**, but switches the mode to **OFF** if the FTP Server generates an error in response.

The **/NONPASV**, **/PASV**, and **/PASV=NEGOTIATE** qualifiers allow you to specify each of the **PASSIVE** mode settings as you start up the FTP Client.

FTP Initialization File

On startup, FTP executes commands in the **FTP.INIT** file in your login directory (if the file exists),

to allow you to customize your FTP sessions. Table 6-4 lists commands you may find useful to have in your FTP.INIT file.

Table 6-4 FTP Commands for the FTP.INIT File

BELL ON	Rings the terminal bell when a file transfer operation is completed.
EXIT-ON-ERROR ON	Causes FTP to exit after any error occurs.
HASH ON	Prints a pound sign (#) for each data buffer transferred.
PROMPT-FOR-MISSING-ARGUMENTS OFF	Disables FTP prompting for missing command line arguments.
PROMPT-ON-CONNECT ON	Automatically prompts for user name and password when a connection to the remote system is established.
SET <i>host</i> /USERNAME: <i>username</i> [/PASSWORD: <i>password</i>]	Sets the default user name or default user name and password for the specified host. If you place SET commands containing passwords in your FTP.INIT file, <i>be careful to protect the file from access by others.</i>
STATISTICS ON	Upon completion of file transfers, displays transfer timing statistics.
VERBOSE ON	Displays all responses from the remote FTP server as they are received.

If you invoke FTP with the /NOINITIALIZATION qualifier, the FTP.INIT file is not processed.

The commands in Table 6-4 are more completely documented in Appendix B.

Troubleshooting FTP

As the first step in any FTP troubleshooting, check the FTP_SERVER_LOG file for error messages.

General Troubleshooting Tips

If the logged information does not help, check the following:

1	Make sure the FTP server is running on the remote system.
2	Ping the FTP server to make sure it is available through the network.
3	If the remote host is on the other side of a firewall, try Passive Mode.
4	Make sure you entered the correct user name and password for the remote system.

Transmitted Files Are Corrupt

If you can copy files, but the files are corrupted after transmission, verify that you are using the correct transfer mode-ASCII or binary. Use ASCII mode for text files and binary mode for executable files, compressed files, graphics files, and any other non-text files. Use Logical-Byte mode if the remote system does not use the standard 8-bit byte.

Copying Files Using TFTP

Like the FTP, TFTP copies files between your system and a remote host. Unlike FTP, you cannot perform operations other than copying files between your system and a remote one (you cannot list directories, delete files, and so on). Also, TFTP does not perform any authentication when transferring files, so a user name and password on the remote host are not required. In general, only files with world read (W:R) access in certain directories on the remote host are available for reading, and only certain directories are available for writing.

Note! TFTP does not check the permissions of directories before attempting to access them. Because the TFTP protocol does not specify any user login or validation, the remote system will probably have some sort of file-access restrictions. The exact restrictions are site-specific and thus cannot be documented here.

The mail option of TFTP, as defined in RFC-783, is obsolete and not supported under the MultiNet TFTP server.

Requirements for TFTP

When you copy a file from a remote host, it must be world-readable (W:R). When copying a file to a remote host:

- A file of the same name must already exist on the remote host.
- The file must be world-writable (W:W).

If these two conditions are not met, TFTP will fail.

Using TFTP

To start TFTP, enter the following command:

```
$ tftp remote_host  
tftp>
```

remote_host is the name of the remote system with which you want to transfer files.

To transfer a file from your system to a remote host, enter a TFTP command in the following format:

```
tftp>put local_file remote_file
```

<i>local_file</i>	Identifies the file you are transferring.
-------------------	---

<i>remote_file</i>	Specifies the name you want the file to have on the remote system. If you specify a file name, it must be an absolute path name (device, directory, and file name). If you do not specify a file name, it defaults to the same name as <i>local_file</i> .
--------------------	--

For example, suppose you want to transfer the file "user:[boojum]accts.log" from your system to the file "/x/boojum/accts.log" on the remote host sales.flowers.com. To do this, you would enter the following commands:

```
$ tftp sales.flowers.com
tftp>put user:[boojum]accts.log /x/boojum/accts.log
```

Both the directory "/x/boojum" and the file "accts.log" must already exist on the remote host, and "accts.log" must be world-writable.

To transfer a file to your system from a remote host, issue a TFTP command in the following format:

```
$ tftp sales.flowers.com
tftp>get remote_file local_file
```

<i>local_file</i>	Specifies the name you want the file to have on your system. If you do not specify a file name, it defaults to the same name as the <i>remote_file</i> .
<i>remote_file</i>	Identifies the file you want to transfer from the remote host. You must supply an absolute path name (device, directory, and file name).

For example, suppose you want to transfer the file "/x/boojum/accts.log" from the remote host "sales.flowers.com" to the file "user:[boojum]accts.log" on the your system. To do this, you would enter the following commands:

```
$ tftp sales.flowers.com
tftp>get /x/boojum/accts.log user:[boojum]accts.log
```

The file "/x/boojum/accts.log" must be world-readable.

Chapter 7

Using DECwindows with MultiNet

Starting with V5.3, OpenVMS supports running DECwindows applications over TCP/IP. This feature provides the ability to run X Windows applications not only between OpenVMS and ULTRIX systems, but also using non-Compaq computer systems that support X Windows (for example, UNIX workstations, Apple Macintosh systems, PCs, and so on). For more information about running DECwindows applications over a network, see the *VMS DECwindows User's Guide*.

For information about Running DECwindows applications over MultiNet TCP/IP see the *Running DECwindows Applications* section.

For information about Authorizing remote systems to access the local display see the *Authorizing Remote Systems* section.

Running DECwindows Applications

To run a DECwindows application on an OpenVMS system over TCP/IP using MultiNet, you must first use the DCL command SET DISPLAY to indicate to DECwindows which system display it should use for the application's user interface.

Note! If you are accessing a remote system using TELNET, RLOGIN, or RSHELL, SET DISPLAY is performed automatically.

Use the /NODE qualifier to specify the remote host name or IP address, and the /TRANSPORT qualifier to specify "TCPIP" transport. The following example shows how to run the application SYS\$SYSTEM:DECW\$PUZZLE.EXE on the local OpenVMS system, and direct the output to an ULTRIX host named ZEPHYR.FLOWERS.COM.

```
$ SET DISPLAY /CREATE /NODE=ZEPHYR.FLOWERS.COM /TRANSPORT=TCPIP
$ RUN SYS$SYSTEM:DECW$PUZZLE
```

Authorizing Remote Systems

Before running a DECwindows application on a remote system and directing the user interface to an OpenVMS workstation running MultiNet, you must authorize the remote system to have access to the local display. Under the DECwindows Session Manager Customize menu, select the Security option. When the Customize Security dialog box appears, specify TCPIP for the Transport, the Internet host name of the remote host for the Node, and a question mark (?) for the Username for each host you wish to grant access to the local display.

Note! EACH user on a workstation who wishes to allow access to the local display from a remote system must specify the remote system under the Customize Security dialog box. A different list is maintained for each user.

Chapter 8

Accessing Remote Systems with the Secure Shell (SSH) Utilities

This chapter describes how to configure and maintain the MultiNet Secure Shell (SSH) client. This is the client side of the software that allows secure interactive connections to other computers in the manner of rlogin/rshell/telnet.

Secure Shell Client (remote login program)

SSH (Secure Shell) is a program for logging into and executing commands on a remote system. It replaces rlogin and rsh, and provides secure encrypted communications between two untrusted hosts over an insecure network. X11 connections and arbitrary TCP/IP ports can be forwarded over the secure channel. SSH connects and logs into the specified hostname. The user must prove his/her identity to the remote system using one of several methods.

First authentication method

If the system the user logs in from is listed in MULTINET:HOSTS.EQUIV or MULTINET:SHOST.EQUIV file on the remote system and the usernames are the same on both sides, the user is permitted to log in.

Second authentication method

If RHOSTS or SHOSTS exists in the user's LOGIN directory on the remote system and contains a line containing the name of the client system and the name of the user on that system, the user is permitted to log in.

This form of authentication alone is not allowed by the server because it is not secure. The second (and primary) authentication method is the RHOSTS or HOSTS.EQUIV method combined with RSA-based host authentication. It means that if the login would be permitted by .RHOSTS, .SHOSTS, MULTINET:HOSTS.EQUIV, or MULTINET:SHOSTS.EQUIV file, and if the client's host key can be verified (see SYS\$DISK:[<login_dir>].SSH]KNOWN_HOSTS and MULTINET:SSH_KNOWN_HOSTS in the FILES section), only then is login permitted. This authentication method closes security holes due to IP spoofing, DNS spoofing, and routing

spoofing.

Note! To the administrator: MULTINET:HOSTS.EQUIV, .RHOSTS, and the rlogin/rshell protocol are inherently insecure and should be disabled if security is desired.

Third authentication method

SSH supports RSA-based authentication. The scheme is based on public-key cryptography. There are cryptosystems where encryption and decryption are done using separate keys, and it is not possible to derive the decryption key from the encryption key.

RSA is one such system. The idea is that each user creates a public/private key pair for authentication purposes. The server knows the public key (SYSSDISK:[<login_dir>.SSH]AUTHORIZED_KEYS lists the public keys permitted for logging), and only the user knows the private key.

When the user logs in:

- 1 The SSH program tells the server the key pair it would like to use for authentication.
- 2 The server checks if this key pair is permitted.
If it is permitted, the server sends the SSH program running on behalf of the user a challenge (a random number) encrypted by the user's public key. The challenge can only be decrypted using the proper private key.
- 3 The user's client then decrypts the challenge using the private key, proving that he/she knows the private key but without disclosing it to the server.
- 4 SSH implements the RSA authentication protocol automatically.

The Key Identity files are created with SSHKEYGEN. To create the RSA key pair files with MultiNet:

- 1 Run SSHKEYGEN to create the RSA key pair: IDENTITY and IDENTITY.PUB.
Both of these files are stored in the user's SYSLOGIN:[.SSH](*directory*). IDENTITY.; is the private key; IDENTITY.PUB is the public key.

Once you have created your identity files:

- 1 Transfer the IDENTITY.PUB file to the remote machine.
- 2 Append the contents of the IDENTITY.PUB file to your AUTHORIZED_KEYS file on the remote machine.
- 3 Update the AUTHORIZED_KEYS file on the remote machine by appending the contents of the public key file to the SYS\$LOGIN:[.SSH]AUTHORIZED_KEYS file on the remote host. The format of the AUTHORIZED_KEYS file requires that each entry consists of a single long line.

After this, the user can log in without giving the password. RSA authentication is much more secure than rhosts authentication. The most convenient way to use RSA authentication may be with an authentication agent. See *sshagent* for more information.

```
$ ! An example of the procedure of setting up MultiNet SSH to enable  
$ ! RSA-based authentication.  
$ ! Using MultiNet SSH client node to connect to a MultiNet SSH server
```

```

node.
$ !
$ ! On the client node
$ !
$ MULTINET SSHKEYGEN
Initializing random number generator...
Generating p: .....++ (distance 662)
Generating q: .....++ (distance 370)
Computing the keys...
Testing the keys...
Key generation complete.
Enter file in which to save the key (DISK$SYS_LOGIN:[DOGBERT.ssh]identity.):
Enter passphrase:
Your identification has been saved in
DISK$SYS_LOGIN:[DOGBERT.ssh]identity..
Your public key is:
1024 33 13428.....29361 DOGBERT@long.hair.com
Your public key has been saved in DISK$SYS_LOGIN:[DOGBERT.ssh]identity.pub
$ !
$ !
$ MULTINET FTP POINTY /USER=DOGBERT/PASSWORD=DEMONSOFTUPIDITY -
_ $ PUT DISK$SYS_LOGIN:[DOGBERT.ssh]identity.PUB -
_ $ DISK$SYS_LOGIN:[DOGBERT.ssh]identity.PUB
long.hair.com MultiNet FTP user process V4.3(119)
Connection opened (Assuming 8-bit connections)
<pointy.hair.com MultiNet FTP Server Process V4.3(16) at Thu 6-Jul-2000
3:20PM-EDT
[Attempting to log in as dogbert]
<User DOGBERT logged into DISK$SYS_LOGIN:[DOGBERT] at Thu 6-Jul-2000
3:21PM-EDT, job 20e00297.
<VMS Store of DISK$SYS_LOGIN:[DOGBERT.SSH]IDENTITY.PUB; started.
<Transfer completed. 395 (8) bytes transferred.
<QUIT command received. Goodbye.
$
$ TELNET POINTY
Trying... Connected to POINTY.HAIR.COM.

```

Authorized Users Only (TM) VAX Operating System, Version V7.1

Username: **DOGBERT**

Password:

Welcome to OpenVMS (TM) VAX Operating System, Version V7.1 on node
POINTY

Last interactive login on Thursday, 6-JUL-2000 08:07

Last non-interactive login on Thursday, 6-JUL-2000 15:20

Logged into POINTY at 6-JUL-2000 15:22:43.68

\$!

\$! For the first entry into the AUTHORIZED_KEYS file copy

```
$ ! (or rename) the file [.SSH]IDENTITY.PUB to [.SSH]AUTHORIZED_KEYS.
$ !
$ COPY [.SSH]IDENTITY.PUB [.SSH]AUTHORIZED_KEYS.
$
$ ! FOR SUBSEQUENT ENTRIES use the APPEND command
$ !
$ APPEND [.SSH]IDENTITY.PUB [.SSH]AUTHORIZED_KEYS.
$
$ ! A sanity check of the file protections shows
$ !
$ DIRECTORY/PROTECTION [.SSH]*.*
```

```
Directory DISK$SYS_LOGIN:[DOGBERT.SSH]
```

```
AUTHORIZED_KEYS.;1      (RWE,RWED,RE,E)
IDENTITY.;1             (RWD,RWD,,)
IDENTITY.PUB;1          (RWE,RWED,RE,E)
KNOWN_HOSTS.;1         (RWD,RWD,,)
RANDOM_SEED.;1          (RWD,RWD,,)
```

```
Total of 5 files.
```

```
$ !
$ DIRECTORY/PROTECTION SSH.DIR
```

```
Directory DISK$SYS_LOGIN:[DOGBERT]
```

```
SSH.DIR;1              (RWD,RWD,,)
```

```
Total of 1 file.
```

Fourth authentication method

If other authentication methods fail, SSH prompts the user for a password.

The password is sent to the remote host for checking. The password cannot be seen on the network because all communications are encrypted. When the server accepts the user's identity it either executes the given command or logs into the system and gives the user a normal shell on the remote system. All communication with the remote command or shell will be encrypted automatically.

The user can disconnect with "~.". All forwarded connections can be listed with "~#". All available escapes can be listed with "~?". A single tilde character can be sent as "~~" (or by following the tilde with a character other than those described above). The escape character must always follow a carriage return to be interpreted as special. The escape character "?" can be changed in configuration files or on the command line.

The session terminates when the command or shell on the remote system exits, or when the user logs out of an interactive session, and all X11 and TCP/IP connections have been closed. The exit status of the remote program is returned as the exit status of SSH. With X11 in use (that is, the DECW\$DISPLAY logical name is set), the connection to the X11 display forwards to the remote side that any X11 programs started from the interactive session (or command) go through the encrypted channel. Also, the connection to the real X server is made from the local system. The

user should not set `DECW$DISPLAY` manually. Forwarding of X11 connections can be configured on the command line or in configuration files.

The `DECW$DISPLAY` value set by SSH points to the server system with a display number greater than zero. This is normal and happens because SSH creates a "proxy" X server on the server system for forwarding the connections over the encrypted channel.

SSH sets up "fake" Xauthority data on the OpenVMS server, as OpenVMS does not support Xauthority currently. It generates a random authorization cookie, stores it in Xauthority on the server, and verifies that any forwarded connections carry this cookie and replace it by the real cookie when the connection is opened. The real authentication cookie is never sent to the server system (and no cookies are sent in the plain). If the user is using an authentication agent, the connection to the agent is forwarded automatically to the remote side unless disabled on the command line or in a configuration file. Forwarding of arbitrary TCP/IP connections over the secure channel can be specified either on the command line or in a configuration file.

One application of TCP/IP forwarding is a secure connection to an electronic purse. Another is going through firewalls. SSH maintains and checks a database containing RSA-based identifications for all hosts it has ever been used with. The database is stored in `SYS$DISK:[<login_dir>.SSH]KNOWN_HOSTS`. Additionally, the file `MULTINET:SSH_KNOWN_HOSTS` is checked for known hosts. Any new hosts are added to the user's file. If a host's identification ever changes, SSH warns about this and disables password authentication to prevent a Trojan horse from getting the user's password. Another purpose of this mechanism is to prevent man-in-the-middle attacks that could be used to circumvent the encryption. The `StrictHostKey-Checking` option (see below) can be used to prevent logins to a system whose host key is not known or has changed.

SSH obtains configuration data from the following sources (in this order):

- 1 command line options
- 2 user's configuration file (`SYS$DISK:[<login_dir>.SSH]CONFIG`)
- 3 system-wide configuration file (`MULTINET:SSH_CONFIG`)

For each parameter, the first obtained value is used. The configuration files contain sections bracketed by "Host" specifications. That section applies only for hosts that match one of the patterns given in the specification. The matched host name is the one given on the command line. Since the first obtained value for each parameter is used, more host-specific declarations should be given near the beginning of the file, and general defaults at the end.

Note! The qualifiers listed in Table 8-1 are position dependent. You must place the qualifier(s) immediately after the SSH command. So the correct syntax is `SSH /qualifier node command`.

Table 8-1 SSH Command Options and Qualifiers

Qualifier	Description
/ALLOW_REMOTE_CONNECT	Allows remote hosts to connect local port forwarding ports. The default is only localhost. May connect to locally binded ports.
/CIPHER= <i>idea</i> <i>des</i> <i>3des</i> <i>blowfish</i> <i>arcfour</i> <i>none</i>	Selects the cipher to use for encrypting the session. is used by default. It is believed to be secure. is the data encryption standard. is encrypt-decrypt-encrypt triple with three different keys. It is more secure than DES. It is used as default if both sites do not support IDEA. is a 128 bit keys encryption algorithm invented by Bruce Schneier. is an algorithm published in the Usenet News in 1995. This algorithm is believed to be equivalent with the RC4 cipher from RSA Data Security (RC4 is a trademark of RSA Data Security). This is the fastest algorithm supported currently. disables encryption entirely. It is intended for debugging only. It renders the connection insecure.
/COMPRESSION	Requests compression of all data (including stdin, stdout, stderr, and data for forwarded X11 and TCP/IP connections). The compression algorithm is the same used by gzip, and the "level" can be controlled by the CompressionLevel option (see below). Compression is desirable on modem lines and other slow connections, but will slow down things only on fast networks. The default value can be set on a host-by-host basis in the configuration files.
/DEBUG	Causes SSH to display debugging messages about its progress. This helps in debugging connection, authentication, and configuration problems. Verbose mode.

Table 8-1 SSH Command Options and Qualifiers (Continued)

Qualifier	Description
/ESCAPE_CHARACTER= <i>ch</i>	<p>Sets the escape character for sessions with a virtual terminal (default: ~). The escape character is recognized only at the beginning of a line. The escape character followed by</p> <ul style="list-style-type: none"> • a dot (.) — closes the connection • a control-Z — suspends the connection • itself — sends the escape character once <p>Setting the character to <i>none</i> disables any escapes and makes the session transparent.</p>
/IDENTITY_FILE= <i>filename</i>	<p>Selects the file from which the identity (private key) for RSA authentication is read. The default is [.SSH]IDENTITY in the user's home directory. Identity files may be specified only on a per-host basis in the configuration file.</p>
/LOCAL_FORWARD= (<i>port:host:hostport</i> ... <i>port:host:hostport</i>)	<p>Causes the given port on the local (client) host to be forwarded to the given host and port on the remote side. The system to which SSH connects acts as the intermediary between the two endpoint systems. Port forwardings can be specified in the configuration file. Only system can forward privileged ports.</p> <p>See the <i>Port Forwarding</i> section for more details.</p>
/LOG_FILE[= <i>logfilename</i>]	<p>Logs all terminal activity to the specified log file. Defaults to SSH.LOG if "<i>logfilename</i>" is not specified.</p>
/NO_AGENT_FORWARDING	<p>Disables forwarding of the authentication agent connection. This may also be specified on a per-host basis in the configuration file.</p>
/OPTION=(" <i>option=value</i> ") /OPTION=(CompressionLevel=6)	<p>Gives options in the format used in the configuration file. This is useful for specifying options for which there is no separate command-line flag. The option has the same format as a line in the configuration file, and are processed prior to any keywords in the configuration file.</p>
/PORT= <i>n</i>	<p>Identifies the port to connect to on the remote host. This can be specified on a per-host basis in the configuration file. The server on the remote host must be listening on the same port for a connection to be established.</p>

Table 8-1 SSH Command Options and Qualifiers (Continued)

Qualifier	Description
/QUIET	Quiet Mode. Causes all warning and diagnostic messages to be suppressed. Only fatal errors display.
/REMOTE_FORWARD= <i>(port:host:hostport</i> ... <i>port:host:hostport)</i>	Causes the given port on the system to which SSH connects to be forwarded to the given host and port on the local side. The system on which the client is running becomes the intermediary between the other two systems. Port forwardings can be specified in the configuration file. Privileged ports can be forwarded only when logging in as system on the remote system. See the <i>Port Forwarding</i> section for more details.
/USE_NONPRIV_PORT	Uses a non-privileged port. With this you cannot use rhosts or rsarhosts authentication, but it can be used to bypass some firewalls that do not allow privileged source ports to pass.
/USERNAME= <i>user</i>	Specifies the name to use to log in as on the remote system. This may be specified on a per-host basis in the configuration file.
/VERSION	Prints the version number of the SSH server only and exits.

Port Forwarding

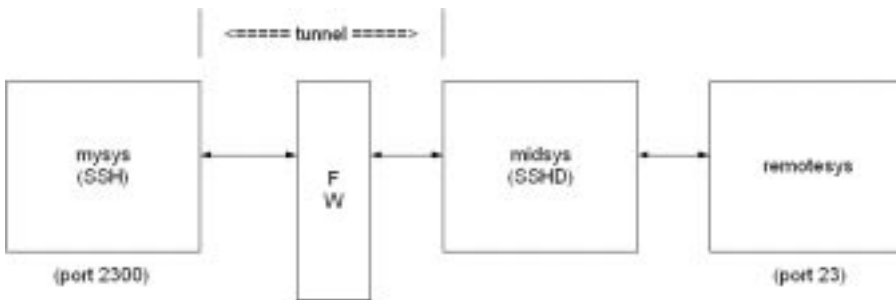
Port forwarding is a mechanism whereby programs that use known TCP/IP ports can have encrypted data forwarded over unsecure connections. This is known as "tunneling" also.

Note! Forwarded ports (tunnels) are only around as long as the SSH session that established them exist; if the SSH session goes away, so do the forwardings.

```
/LOCAL_FORWARD=(localport:remotehost:remoteport)
```

This causes `localport` on the system the client is running on to be forwarded to `remotehost:remoteport`. The system to which SSH connects acts as the intermediary between the two endpoint systems.

For example: Use port forwarding to allow a system (`midsys`) to encrypt and forward TELNET sessions between itself (`mysys`) that's outside a corporate firewall to a system (`remotesys`) that is inside a corporate firewall. Note that the use of port 2300 in the examples is arbitrary.



On the SSH command line from mysys:

```
$ ssh midsys /local_forward=(2300:remotesys:23)
```

With the SSH session to midsys now active, type in another window on mysys:

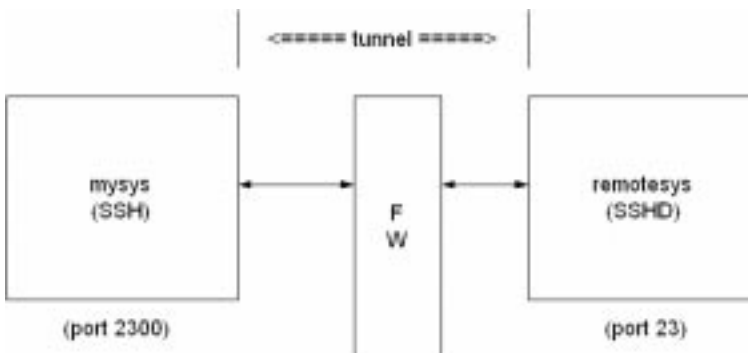
```
$ telnet localhost /port=2300
```

Note! The SSH session must remain active for port forwarding activity.

This causes a connection to mysys : 2300. The SSH client has bound to this port, and will see the connection request. SSH sends an "open channel" request to midsys, telling it there's a connect request for port 23 on remotesys. Midsys will connect to remotesys : 23, and send back the port information to mysys. Mysys completes the connection request, and the TELNET session between mysys and remotesys is now in place, using the tunnel just created through the firewall between mysys and midsys.

All traffic between mysys and midsys (through the firewall) is encrypted/decrypted by SSH on mysys and SSHD on midsys, and hence, is safe. TELNET does not know this, of course, and does not care.

Note that ports can also be forwarded from a localhost to the remotehost that's running SSHD, as illustrated in this figure.



In this example, port 2300 on mysys is being forwarded to remotesys : 23. To do this, use SSH

on `mysys`:

```
$ ssh remotesys /local_forward=(2300:remotesys:23)
```

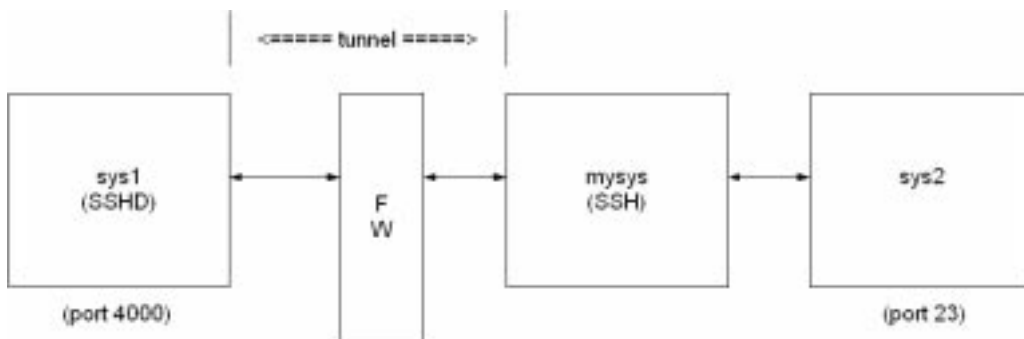
Then, also on `mysys`, type:

```
$ telnet localhost /port=2300
```

When SSH and SSHD start their dialog, SSHD on `remotesys` connects back to itself, port 23, and the TELNET session is established.

```
/REMOTE_FORWARD=(remoteport1:remotehost:remoteport2)
```

This causes `remoteport1` on the system to which SSH connects to be forwarded to `remotehost:remoteport2`. In this case, the system on which the client is running becomes the intermediary between the other two systems.



For example, say I want to use `mysys` to create a tunnel between `sys1:4000` and `sys2:23`, so that TELNET sessions that originate on `sys1:4000` get tunneled to `sys2` through the firewall.

On `mysys`:

```
$ ssh sys1 /remote_forward=(4000:sys2:23)
```

Now, on `sys1`, a user could establish a TELNET session to `sys1` by doing:

```
$ telnet localhost /port=4000
```

The mechanism used for making the TELNET connection (setting up the tunnel) is essentially the same as described in the `/LOCAL_FORWARD` example above, except that the roles of SSH and SSHD in the dialog are reversed.

CONFIGURATION FILES

The configuration file has the following format: empty lines and lines starting with `#` are comments. Otherwise, a line is of the format "keyword arguments" or "keyword =arguments". The possible keywords and their meanings are as follows:

Note! The configuration files are case-sensitive, but keywords are case-insensitive:

Table 8-2 Configuration File Keywords

Keyword	Description
BatchMode	Disables passphrase/password querying if set to "yes". Use this option in scripts and other batch jobs where you have no user to supply the password. The argument must be "yes" or "no". The default is no.
Cipher	Specifies the cipher to use for encrypting the session. Currently, idea, des, 3des, blowfish, arcfour, and none are supported. The default is "idea" (or "3des" if "idea" is not supported by both hosts). Using "none" (no encryption) is intended only for debugging and renders the connection insecure.
ClearAllForwardings	Clears all forwardings after reading all config files and parsing the command line. This disables forwardings in the config file when you want to make a second connection to the host having forwardings in the config file. By default, SCP sets this on so it will not fail even if you have some forwardings set in the config file.
Compression	Specifies whether to use compression. The argument must be "yes" or "no". The default is no.
CompressionLevel	Specifies the compression level to use if compression is enabled. The argument must be an integer from 1 (fast) to 9 (slow, best). The default level is 6, which is good for most applications. The meaning of the values is the same as in GNU GZIP.
ConnectionAttempts	Specifies the number of tries (one per second) to make before falling back to rsh or exiting. The argument must be an integer. This may be useful in scripts if the connection sometimes fails. The default is 4.
EscapeChar	Sets the escape character (default: ~). The argument should be a single character, '^' followed by a letter, or "none" to disable the escape character entirely (making the connection transparent for binary data).

Table 8-2 Configuration File Keywords (Continued)

Keyword	Description
FallbackToRsh	Specifies that if connecting via SSH fails due to a connection refused error (there is no SSHD listening on the remote host), rsh should be used instead (after a suitable warning about the session being unencrypted). The argument must be "yes" or "no". The default is no.
ForwardAgent	Specifies whether the connection to the authentication agent (if any) will be forwarded to the remote system. The argument must be "yes" or "no". The default is yes.
ForwardX11	Specifies whether X11 connections will be redirected over the secure channel and DECW\$DISPLAY set. The argument must be "yes" or "no". The default is 0.
GatewayPorts	Specifies that remote hosts may connect locally to forwarded ports. The argument must be "yes" or "no".
GlobalKnownHostsFile	Specifies a file to use instead of MULTINET:SSH_KNOWN_HOSTS.
Host	Restricts the following declarations (up to the next Host keyword) to be only for those hosts that match one of the patterns given after the keyword. * and ? can be wildcards in the patterns. A single * as a pattern can be used to provide global defaults for all hosts. The host is the hostname argument given on the command line (that is, the name is not converted to a fully-qualified host name before matching).
IdentityFile	Specifies the file from which the user's RSA authentication identity is read (the default being [.SSH]IDENTITY in the user's home directory). Any identities represented by the authentication agent are used for authentication. It is possible to have multiple identity files specified in configuration files; all these identities will be tried in sequence. The default is Identity. in the user's [.SSH] directory.

Table 8-2 Configuration File Keywords (Continued)

Keyword	Description
KeepAlives	Specifies whether the system should send keepalive messages to the other side. If they are sent, death of the connection or crash of one of the systems will be noticed. This means connections will die if the route is down temporarily. The default is yes (to send keepalives), and the client will notice if the network goes down or the remote host dies. This is important in scripts. To disable keepalives, the value should be set to "no" in both the server and the client configuration files.
LocalForward	Specifies that a TCP/IP port on the local system be forwarded over the secure channel to given <i>host:port</i> from the remote system. The first argument must be a port number, and the second must be <i>host:port</i> . Multiple forwardings may be specified, and additional forwardings can be given on the command line. Only the system can forward privileged ports.
NumberOfPasswordPrompts	<p>Specifies the number of password prompts before giving up. The argument must be an integer.</p> <p>Note! The server limits the number of attempts (currently 5). Setting this number larger has no effect. The default value is one.</p> <p>The default is 1.</p>
PasswordAuthentication	Specifies whether to use password authentication. The argument to this keyword must be "yes" or "no". The default is yes.
PasswordPromptHost	Specifies whether to include the remote host name in the password prompt. The argument to this keyword must be "yes" or "no".
PasswordPromptLogin	Specifies whether to include the remote login name in the password prompt. The argument to this keyword must be "yes" or "no". The default is yes.
Port	Specifies the port number to connect on the remote host. The default is 22.

Table 8-2 Configuration File Keywords (Continued)

Keyword	Description
ProxyCommand	Specifies the command to use to connect to the server. The command string extends to the end of the line. In the command string, <i>%h</i> is substituted by the host name to connect and <i>%p</i> is substituted by the port. The command can be anything, and should read from its stdin and write to its stdout. It should connect an SSHD server running on some system. Host key management will be done using the HostName of the host being connected (defaulting to the name typed by the user).
RemoteForward	Specifies that a TCP/IP port on the remote system be forwarded over the secure channel to given <i>host:port</i> from the local system. The first argument must be a port number, and the second must be <i>host:port</i> . Multiple forwardings may be specified, and additional forwardings can be given on the command line. Only the SYSTEM can forward privileged ports.
RhostsAuthentication	<p>Specifies whether to try rhosts-based authentication.</p> <p>Note! This declaration affects the client side only and has no effect on security.</p> <p>Disabling rhosts authentication may reduce authentication time on slow connections when rhosts authentication is not used. Most servers do not permit RhostsAuthentication because it is not secure (see RhostsRSAAuthentication). The argument must be "yes" or "no". The default is yes.</p>
RhostsRSAAuthentication	Specifies whether to try rhosts-based authentication with RSA host authentication. This is the primary authentication method for most sites. The argument must be "yes" or "no". The default is yes.
RSAAuthentication	Specifies whether to try RSA authentication. The argument must be "yes" or "no". RSA authentication will be attempted only if the identity file exists, or an authentication agent is running. The default is yes.

Table 8-2 Configuration File Keywords (Continued)

Keyword	Description
StrictHostKeyChecking	If this is set to "yes", SSH will never add host keys to the <code>[.SSH]KNOWN_HOSTS</code> file in <code>SYSS\$LOGIN</code> : automatically. It refuses to connect hosts whose host key has changed also. This provides maximum protection against trojan horse attacks. However, it can be somewhat annoying if you don't have good <code>MULTINET:SSH_KNOWN_HOSTS</code> files installed and frequently connect new hosts. This option forces the user to add manually any new hosts. Normally, this option is set to "ask", and new hosts will be added automatically to the known host files after you have confirmed you want to do that. If this is set to "no", a new host will be added to the known host files automatically. The host keys of known hosts will be verified automatically in either case. The argument must be "yes", "no", or "ask". The default is yes.
UsePrivilegedPort	Specifies whether to use privileged port when connecting to the other end. The default is yes if <code>rhhosts</code> or <code>rsarhosts</code> authentications are enabled. The user specifies the user to log in as. This can be useful if you have a different user name on different systems. This saves the trouble of having to remember to give the user name on the command line. The default is yes.
UserKnownHostsFile	Specifies a file to use instead of <code>[.SSH]KNOWN_HOSTS</code> in <code>SYSS\$LOGIN</code> .
UseRsh	Specifies that <code>rlogin/rshell</code> should be used for this host. It is possible that the host does not support the SSH protocol. This causes SSH to execute <code>rsh</code> . All other options (except <code>Host-Name</code>) are ignored if this has been specified. The argument must be "yes" or "no".

Other Files

The following files are used by SSH. Note that these files reside generally in the `[.SSH]` subdirectory from the user's `SYSS$LOGIN` directory. The `[.SSH]` subdirectory is created automatically on your local system the first time SSH is executed, and on a remote OpenVMS system the first time an SSH connection is made to that system. File protection for `SYSS$LOGIN:SSH.DIR` should be (S:RWD, O:RWD, G:R, W).

Table 8-3 SSH Files

File Name	Resides On	Description
[.SSH]AUTHORIZED_KEYS	Server System	Lists the RSA keys that can be used for logging in as this user. The format is the same as the IDENTITY.PUB files; that is, each line contains the number of bits in modulus, public exponent, modulus, and comment fields, separated by spaces. This file is not sensitive. The recommended permissions are (S:RWD,O:RWD,G:;W:), and it must be owned by the user.
[.SSH]CONFIG.	Client System	This is the per-user configuration file. This file is used by the SSH client. It does not contain sensitive information. The recommended file protection is (S:RWD,O:RWD,G:;W:).
[.SSH]IDENTITY.	Client System	Contains the RSA authentication identity of the user. This file is generated by SSHAGENT and contains sensitive data, and MUST have a file protection of (S:RWD,O:RWD,G:;W:), and it must be owned by the user. It is possible to specify a passphrase when generating the key. The passphrase is used to encrypt the sensitive part of this file using IDEA.
[.SSH]IDENTITY.PUB	Client System and Server System	Contains the public key for authentication. This is the public part of the identity file in readable format. This file should be added to [.SSH]AUTHORIZED_KEYS on all systems where you want to log in using RSA authentication. This file is not sensitive and can, but need not be, readable by anyone. This file is never used automatically and is not necessary; it is provided for the convenience of the user only.

Table 8-3 SSH Files (Continued)

File Name	Resides On	Description
[.SSH]KNOWN_HOSTS	Client System	Records host keys for all hosts the user has logged into that are not in MULTINET:SSH_KNOWN_HOSTS.
[.SSH]RANDOM_SEED.	Client System	<p>Seeds the random number generator. This file contains sensitive data and MUST have a protection of no more than (S:RWD,O:RWD,G:,W:), and it must be owned by the user. This file is created the first time the program is run and is updated automatically. The user should never need to read or modify this file. On OpenVMS systems, multiple versions of this file will be created; however, all older versions of the file may be safely purged.</p> <p>Use the DCL command: SET FILE /VERSION_LIMIT=n RANDOM_SEED to set a limit on the maximum number of versions of this file that may exist at any given time.</p>

Table 8-3 SSH Files (Continued)

File Name	Resides On	Description
.RHOSTS	Server System	<p>Is used in rhosts authentication to list the host/user pairs that are permitted to log in.</p> <p>Note! This file is also used by rlogin and rshell, which makes using this file insecure.</p> <p>Each line of the file contains a host name (in the fully-qualified form returned by name servers), and then a user name on that host, separated by a space. This file must be owned by the user, and must not have write permissions for anyone else. The recommended permission is read/write for the user, and not accessible by others.</p> <p>Note! By default SSHD is installed so that it requires successful RSA host authentication before permitting rhosts authentication. If your server system does not have the client's host key in the file MULTINET:SSH_KNOWN_HOSTS, you can store it in SYS\$LOGIN:SSH_KNOWN_HOSTS. The easiest way to do this is to connect back to the client from the server system using SSH; this will add the host key in [.SSH]KNOWN_HOSTS in SYS\$LOGIN: automatically.</p>
.SHOSTS	Server System	<p>Is used the same way as .RHOSTS. The purpose for having this file is to be able to use rhosts authentication with SSH without permitting login with rlogin or rshell.</p>
MULTINET:HOSTS.EQUIV	Server System	<p>Is used during .rhosts authentication. It contains fully-qualified hosts names, one per line. If the client host is found in this file, login is permitted provided client and server user names are the same. Additionally, successful RSA host authentication is required. This file should only be writeable by SYSTEM.</p>

Table 8-3 SSH Files (Continued)

File Name	Resides On	Description
MULTINET:SHOSTS.EQUIV	Server System	Is processed exactly as <code>MULTINET:HOSTS.EQUIV</code> . This file may be useful to permit logins using SSH but not using <code>rshell/rlogin</code> .
MULTINET:SSH_CONFIG	Client System	This is a system-wide configuration file. This file provides defaults for those values that are not specified in the user's configuration file, and for those who do not have a configuration file. This file must be world-readable.
MULTINET:SSH_KNOWN_HOSTS	Server System	<p>Is a system-wide list of known host keys. This file should be prepared by the system administrator to contain the public host keys of all systems in the organization. It should be world-readable and contain public keys, one per line, in the following format fields, separated by spaces: system name, number of bits in modulus, public exponent, modulus, and optional comment field.</p> <p>When different names are used for the same system, all such names should be listed, separated by commas. The fully-qualified system name (as returned by name servers) is used by SSHD to verify the client host when logging in. Other names are needed because SSHD does not convert the user-supplied name to a fully-qualified name before checking the key, because someone with access to the name servers would then be able to fool host authentication.</p>

SSHAgent (authentication agent)

multinet sshagent command

DESCRIPTION

SSHAGENT is a program that holds authentication private keys. The SSHAGENT is started in the beginning of an X-session or a login session, and all other windows or programs are started as children of the SSHAGENT program. The command normally starts X or is the user shell.

Programs started under the agent inherit a connection to the agent. The agent is used for RSA authentication when logging to other systems using SSH. If the agent is started without any arguments (no command), it starts as a background process. The command sets the SSH_AUTH_SOCK and SSH_AGENT_PID logical names. The command sets the SSH_AGENT_username_MBX and SSH_AGENT_username_PID logical names.

The agent does not have any private keys initially. Keys are added using SSHADD. When executed without arguments, SSHADD adds the [.SSH]IDENTITY file from SYS\$LOGIN:. If the identity has a passphrase, SSHADD asks for the passphrase. It then sends the identity to the agent. Several identities can be stored in the agent; the agent can use any of these identities automatically. SSHADD -l displays the identities currently held by the agent. The idea is that the agent is run in the user's workstation. However, it can be run on a shared system as well.

Authentication data need not be stored on any other system. Authentication passphrases never go over the network. The connection to the agent is forwarded over SSH remote logins. The user can use the privileges given by the identities anywhere in the network in a secure way.

A connection to the agent is available to all programs run by the user. The names of the mailboxes used are stored in the SSH_TO_AGENT username-MBX and SSH_FROM_AGENT username-MBX AUTH_SOCK environment variable. The mailboxes are accessible only to the current user. This method is easily abused by SYSTEM or by another instance of the same user.

FILES

[.SSH]IDENTITY in SYS\$LOGIN:	Contains the RSA authentication identity of the user. This file should not be readable by anyone but the user. It is possible to specify a passphrase when generating the key. That passphrase is used to encrypt the private part of this file. This file is not used by SSHAGENT, but is added to the agent using SSHADD at login.
----------------------------------	--

SSHADD

Adds identities for the authentication agent.

```
multinet sshadd [/LIST] [/DELETE] [/PURGE] [file...]
```

DESCRIPTION

SSHADD adds identities to SSHAGENT, the authentication agent. When run without arguments, SSHADD adds the file [.SSH]IDENTITY file from SYS\$LOGIN:. Alternative file names can be given on the command line. If any file requires a passphrase, SSHADD asks for the passphrase from the user.

The authentication agent must be running and must have been executed by the user for SSHADD to work.

OPTIONS

/DELETE	Instead of adding the identity, removes the identity from the agent.
/PURGE	Deletes all identities from the agent.
/LIST	Lists all identities currently represented by the agent.

RETURN STATUS

M	<p>SSHADD returns one of the following exit statuses. These may be useful in scripts.</p> <p>0—The requested operation was performed successfully.</p> <p>1—No connection could be made to the authentication agent. Presumably there is no authentication agent active in the execution environment of SSHADD.</p> <p>2—The user did not supply a required passphrase.</p> <p>3—An identify file could not be found, was not readable, or was in bad format.</p> <p>4—The agent does not have the requested identity.</p> <p>5—An unspecified error has occurred; this is a catch-all for errors not listed above.</p>
---	---

FILES

[.SSH]IDENTITY in SYS\$LOGIN:	<p>Contains the RSA authentication identity of the user. This file should not be readable by anyone but the user. It is possible to specify a passphrase when generating the key. That passphrase is used to encrypt the private part of this file. This is the default file added by SSHADD when no other files have been specified.</p> <p>If SSHADD needs a passphrase, it reads the passphrase from the current terminal if it was run from a terminal. If SSHADD does not have a terminal associated with it but DECW\$DISPLAY is set, it opens an X11 window to read the passphrase.</p>
----------------------------------	--

SSHKEYGEN

Generates authentication key pairing.

```
multinet sshkeygen [/BITS=n] [/IDENTITY_FILE=file]
                  [/PASSPHRASE=passphrase] [/COMMENT=comment]
multinet sshkeygen /CHANGE_PASSPHRASE [/PASSPHRASE=old_passphrase]
                  [/NEW_PASSPHRASE=new_passphrase]
multinet sshkeygen /CHANGE_COMMENT [/PASSPHRASE=passphrase]
                  [/COMMENT=comment]
multinet sshkeygen /CHANGE_CIPHER [/IDENTITY_FILE=file]
                  [/PASSPHRASE=passphrase]
multinet sshkeygen [/HOST][/BITS=n][/COMMENT=comment]
```

DESCRIPTION

SSHKEYGEN generates and manages authentication keys for SSH. Each user wanting to use SSH with RSA authentication runs SSHKEYGEN once to create the authentication key in SYS\$LOGIN:[.SSH]IDENTITY. The system administrator may use this to generate host keys. This program generates the key and asks for a file in which to store the private key. The public key is stored in a file with the same name but ".pub" appended. The program asks for a passphrase. The passphrase may be empty to indicate no passphrase (host keys must have empty passphrase), or it may be a string of arbitrary length. Good passphrases are 10-30 characters long and are not simple sentences or otherwise easily guessable. The passphrase can be changed later by using the /CHANGE_PASSPHRASE option.

There is no way to recover a lost passphrase. If the passphrase is lost or forgotten, you need to generate a new key and copy the corresponding public key to other systems.

There is also a comment field in the key file that is only for convenience to the user to help identify the key. The comment can tell what the key is for, or whatever is useful. The comment is initialized to user@host when the key is created, but can be changed using the /CHANGE_CIPHER option. Using the /CHANGE_CIPHER option, keys encrypted in any supported cipher can be updated to

use the default cipher which is 3DES.

Note! When the `/HOST` qualifier is used, the `/IDENTITY_FILE=file.nam` is ignored.

OPTIONS

<code>/BITS=n</code>	Specifies the number of bits in the key to create. Minimum is 512 bits. Generally 1024 bits is considered sufficient, and key sizes above that no longer improve security but make things slower. The default is 1024 bits.
<code>/CHANGE_CIPHER</code>	Requests that the key's cipher is changed to the current default cipher (determined at compile-time — currently 3DES).
<code>/CHANGE_COMMENT</code>	Requests changing the comment in the private and public key files. The program prompts for the file containing the private keys, for the passphrase if the key has one, and for the new comment.
<code>/CHANGE_PASSPHRASE</code>	Requests changing the passphrase of a private key file instead of creating a new private key. The program prompts for the file containing the private key, for the old passphrase, and twice for the new passphrase.
<code>/COMMENT=<i>comment</i></code>	Provides a comment.
<code>/HOST</code>	Specifies that the host key is being generated. When this option is specified, there is no prompt for passphrases, and the key file defaults to MULTINET_ROOT:[MULTINET]SSH_HOST_KEY.
<code>/IDENTITY_FILE=<i>file</i></code>	Specifies the file name in which to load/store the key.
<code>/NEW_PASSPHRASE=<i>passphrase</i></code>	Provides the new passphrase.
<code>/PASSPHRASE=<i>passphrase</i></code>	Provides the current passphrase. If you are generating a key file for use as a host key file without using the <code>/HOST</code> option, do not include a passphrase; the server will not start if it encounters one.

FILES

These files exist in SYS\$LOGIN:

[.SSH]IDENTITY.	Contains the RSA authentication identity of the user. This file should not be readable by anyone but the user. It is possible to specify a passphrase when generating the key; that passphrase will be used to encrypt the private part of this file using 3DES. This file is not accessed automatically by SSHKEYGEN, but it is offered as the default file for the private key.
[.SSH]IDENTITY.PUB	Contains the public key for authentication. The contents of this file should be added to [.SSH]AUTHORIZED_KEYS on all systems where you want to log in using RSA authentication. There is no need to keep the contents of this file secret.
[.SSH]RANDOM_SEED	Seeds the random number generator. This file should not be readable by anyone but the user. This file is created the first time the program is run, and is updated every time SSHKEYGEN is run.

Appendix A

DCL User Commands

This appendix lists the commands you can invoke from the DCL command line.

Command Summary

The following table lists the MultiNet user DCL commands:

Table A-1 DCL Command Summary

Utility	Description
MULTINET DECODE	Decodes a file encoded by the MultiNet SMTP mail handler.
MULTINET FINGER	Displays information about users currently logged into local or remote systems.
MULTINET FTP	Uses the standard Internet FTP protocol to transfer files between TCP/IP hosts, and allows you to manipulate them.
MULTINET KERBEROS DESTROY	Deletes Kerberos authentication tickets you previously acquired.
MULTINET KERBEROS INIT	Acquires the initial ticket that allows client programs to obtain tickets to access network services.
MULTINET KERBEROS LIST	Displays your ticket status.
MULTINET KERBEROS PASSWORD	Changes your Kerberos password.
MULTINET LPRM	Cancels print jobs, specified by job number, from the SYSS\$PRINT queue.

Table A-1 DCL Command Summary (Continued)

Utility	Description
MULTINET RCP	Transfers file between TCP/IP hosts.
MULTINET REMIND	Creates reminders to be sent at specified intervals by either mail or broadcast to the recipient's terminal.
MULTINET RLOGIN	Connects your terminal to another system on the network.
MULTINET RSHELL	Runs commands on a remote system and displays the command output on your terminal.
MULTINET RUSERS	Displays information about users logged into local or remote systems.
MULTINET SEND	Sends a brief message to another user's terminal.
MULTINET TALK	Initiates an interactive conversation with another user on the local host or on any remote host that supports the TALK protocol.
MULTINET TELNET	Logs into a remote host from the local host.
MULTINET TFTP	Transfers files between TCP/IP hosts.
MULTINET WHOIS	Displays information about users registered with the Internet Network Information Center (InterNIC).

MULTINET DECODE

Decodes a file encoded by the MultiNet SMTP mail handler.

FORMAT

MULTINET DECODE *input_file output_file*

PARAMETERS

input_file

Specifies the name of a file containing the encoded file, including the RFC822 headers at the top of the message. The message must include MIME-Version, Content-Type, and Content-Transfer-Encoding headers in order to be decoded. Only the APPLICATION/RMS content-type and base64 content-transfer-encoding are supported.

output_file

The name for the resulting decoded file.

EXAMPLE

Binary files can be sent via SMTP using the undocumented /FOREIGN qualifier of the OpenVMS Mail SEND command. The following example shows how to send such a file and use DECODE to translate the corresponding mail message:

1	First, send a executable file using OpenVMS Mail: \$ MAIL MAIL> SEND /FOREIGN /NOEDIT BINARY.EXE To: SMTP%"TREEFROG@ABC.COM" Subj: BINARY.EXE
2	When the file arrives, store the ASCII-encoded mail as a text file: \$ MAIL MAIL> EXTRACT/NOHEADER BINARY.TXT
3	Finally, decode the BINARY.TXT file into an executable file: \$ MULTINET DECODE BINARY.TXT BINARY.EXE

MULTINET FINGER

Displays information about users currently logged into local or remote systems.

FORMAT

MULTINET FINGER [*user_name*] [*@host_name*]

PARAMETERS

user_name

Specifies the user name about which to obtain detailed information. If not specified, brief information is displayed about users currently logged in.

host_name

The name (or network address) of the host to which a connection should be made. If you don't specify a host name, information about the local host is displayed. The host name can be specified as an IP address; for example: \$ **MULTINET FINGER @127.0.0.1**

QUALIFIERS

/NOCLUSTER

Restricts output to that of a single system instead of its VMScluster.

/CLUSTER

Displays all cluster users.

Restrictions

To display information about users logged into a remote system, that system must have a FINGER server enabled.

EXAMPLE

```
$ MULTINET FINGER
Friday, April 7,2000 12:39AM-PDT Up 0 02:10:27 4+0 Load ave 0.24 0.25 0.19
User   Personal Name   Job   Subsys   TTY   Console Location
SYSTEM System Manager   37    *DCL*   TTA3   Macintosh SE
SMITH  L. Stuart Smith    32    FINGER  FTA1   Console
                      33    *DCL*   FTA2   Console
                      35    *DCL*   FTA3   Console
```


MULTINET FTP

Uses the standard Internet FTP protocol to transfer files between TCP/IP hosts, and allows you to manipulate them.

FORMAT

MULTINET FTP [*host*] [*command*]

PARAMETERS

host

Specifies the name of a remote host to which you want to connect. You can also specify the host name as an IP address. If you enter the name of a remote host on the DCL command line, FTP immediately attempts to connect to the FTP server on that host. If you don't specify a remote host, FTP enters its TOPS-20 style command interpreter and prompts for FTP commands.

command

Specifies an FTP command to execute. If you do not specify a command, FTP starts interactive mode and prompts for commands.

Note! You must specify all FTP DCL qualifiers on the command line before any *command*.

If *command* causes an FTP error to occur, the error condition is reported back to DCL in the \$STATUS symbol. To determine if an FTP error occurred, examine the hexadecimal value of \$STATUS. If the lower byte is the value %X2C, the FTP error code can be determined by dropping the high order four bits of the 32-bit condition code and examining the next twelve. For example, if you specify the incorrect remote password, the FTP error status code returned by the server will be the decimal value 530. As the FTP image exits, the error status (and hence the \$STATUS symbol) is set to the value %X1212002C (decimal 530 is the same as hexadecimal %X212).

QUALIFIERS

/ACCOUNT=account_name

Specifies your account name. In addition to a user name and password for validation, some systems require an account string. MultiNet preserves the case of characters placed within quotation marks. Characters not placed within quotation marks are converted to lowercase. Be aware the some systems might not recognize these lowercase characters and deny access.

/BINARY

Equivalent to /TYPE=IMAGE, this qualifier allows you to transfer binary files. You can override the /BINARY qualifier with the TYPE command in interactive mode.

/IMAGE

Equivalent to /TYPE=IMAGE, this qualifier allows you to transfer binary files. You can override the /IMAGE qualifier with the TYPE command in interactive mode.

/INITIALIZATION (default)**/NOINITIALIZATION**

Tells FTP to read commands from your SYS\$LOGIN:FTP.INIT file when invoked. Use the /NOINITIALIZATION qualifier to disable this behavior.

{ STREAM (default) }

/MODE={ COMPRESS }

{ user-defined-mode }

Specifies the file transfer mode. You can change the MODE by using the MODE command in interactive mode, and default to STREAM. A user-defined mode can be created as an executable file.

/PASSWORD=password

Specifies the password to use on the remote host, which must be specified in conjunction with the /USERNAME qualifier. If not specified, FTP prompts for the password. MultiNet preserves the case of characters placed within quotation marks. Characters not placed within quotation marks are converted to lowercase. Be aware the some systems might not recognize these lowercase characters and deny access.

/PORT=port

Specifies an alternate TCP port number to use when connecting to the FTP control port on the remote host. You should only use this qualifier when communicating with an FTP server that uses a non-standard control port number.

{ CONNECT, }

/PROMPT[({ NOMISSING_ARGUMENTS })]

Modifies the operation of FTP. If /PROMPT=CONNECT is used following a successful connection, FTP automatically prompts for a user name and password to send to the remote system. The same result can be achieved by adding the line PROMPT-ON-CONNECT ON to your SYS\$LOGIN:FTP.INIT file.

If you use /PROMPT=NOMISSING_ARGUMENTS, FTP does not prompt you for missing command line arguments. The same behavior can be accomplished by adding the line PROMPT-FOR-MISSING-ARGUMENTS OFF to your SYS\$LOGIN:FTP.INIT file.

For compatibility with previous releases of MultiNet, using the /PROMPT qualifier alone implies /PROMPT=CONNECT.

/STATISTICS**/NOSTATISTICS (default)**

Sets the FTP STATISTICS flag so FTP displays transfer timing statistics upon completion of file transfers.

{ FILE }

/STRUCTURE={ RECORD }

{ VMS }

Specifies the STRUCTURE of the file transfers. You can change the STRUCTURE by using the

STRUCTURE command in interactive mode. The default is FILE, or VMS when communicating between systems running MultiNet. The /STRUCTURE qualifier disables automatic negotiation of VMS structure.

/TAKE_FILE=file

Causes FTP to execute commands from the specified file before entering command mode. This qualifier is functionally equivalent to re-directing SYS\$INPUT:.

{ ASCII }

{ IMAGE }

/TYPE={ BACKUP }

{ LOGICAL_BYTE }

Specifies the file transfer TYPE. You can change the TYPE by using the TYPE command (which defaults to ASCII) in interactive mode.

/USERNAME=username

Specifies the user name to use on the remote host. MultiNet preserves the case of characters placed within quotation marks. Characters not placed within quotation marks are converted to lowercase. Be aware the some systems might not recognize these lowercase characters and deny access.

/VERBOSE

/NOVERBOSE (default)

Sets the FTP VERBOSE flag. Causes FTP to display all responses from the remote FTP server as they are received.

/VMS_STRUCTURE_NEGOTIATION (default)

/NOVMS_STRUCTURE_NEGOTIATION

Causes the FTP client to send a STRU O VMS FTP command to the server FTP to negotiate transparent transfer of files with arbitrary RMS attributes. If the server responds with an error, the default transfer structure of FILE is assumed. The negotiation takes place after a connection has been successfully opened.

You can use the /NOVMS_STRUCTURE_NEGOTIATION qualifier to disable this feature if automatic negotiation causes unforeseen problems with another vendor's server.

EXAMPLES

This example shows how to establish a connection to the host FLOWERS.COM with prompting for a remote user name and password, and printing statistics for the duration of the session (or until the user turns it off).

```
$ FLOWERS.10M /PROMPT=CONNECT /STATISTICS
```

This example shows how to establish a connection to the host DS.INTERNIC.NET, log in with the user name ANONYMOUS and password GUEST, and fetch the file RFC:RFC959.TXT (the FTP Request for Comments), placing it in the file RFC959.TXT in your default directory.

```
$ /USER=ANO2YMOUS /PASSWORD=GUEST DS.INTERNIC.NET -  
_ $ GET RFC:RFC959.TXT RFC959.TXT
```

MULTINET KERBEROS DESTROY

Deletes Kerberos authentication tickets you previously acquired.

FORMAT

MULTINET KERBEROS DESTROY

QUALIFIERS

/QUIET (default)

/NOQUIET

Determines if the terminal bell sounds when tickets cannot be destroyed.

/STATUS (default)

/NOSTATUS

Determines if a message appears when the tickets are destroyed.

EXAMPLE

This example shows how to destroy your tickets.

```
$ MULTINET KERBEROS DESTROY  
Tickets destroyed.  
$
```

MULTINET KERBEROS INIT

Acquires the initial ticket that allows client programs to obtain tickets to access network services.

FORMAT

MULTINET KERBEROS INIT

QUALIFIERS

/INSTANCE="name"

Specifies the instance to use in obtaining the initial ticket (by default, an empty string).

/LIFETIME=minutes

Specifies how long the ticket can be used. The specified value is in minutes and can range from 5 to 1275 (21 hours, 15 minutes). Typically, the default is set to 480 (8 hours). You can change the default by using the MULTINET KERBEROS DATABASE EDIT utility to edit the DEFAULT principal name.

/REALM=realm

Specifies the Kerberos realm to use. The default is the local realm name specified in the MULTINET:KERBEROS.CONFIGURATION file.

Note! The realm name is case-sensitive.

/USERNAME=login_name

Specifies an alternate *login_name*.

/VERBOSE

/NOVERBOSE (default)

Specifies whether displayed messages should provide extra information.

EXAMPLE

```
$ KERBEROS INIT /REALM=FLOWERS.COM
$
```

MULTINET KERBEROS LIST

Displays your ticket status.

FORMAT

MULTINET KERBEROS LIST

QUALIFIERS

/BRIEF

/NOBRIEF (default)

Lists only the acquired tickets without issuance dates, expiration dates, principal name, or the ticket file name.

/CHECK_TGT

/NOCHECK_TGT (default)

Determines if the tickets are still valid and returns an exit status of either success or failure. (TGT stands for ticket-getting ticket.) The default is to indicate ticket status with a message on the screen.

/SRVTAB

Lists the contents of the MULTINET:KERBEROS.SRVTAB file which indicates what services are available. This can provide an administrator with useful information about what services are configured in the Kerberos database.

EXAMPLE

This example shows how to list the ticket status.

```
$ MULTINET KERBEROS LIST
```

```
Principal: john@FLOWERS.COM
```

```
Issued                Expires                Principal
```

```
June 12 16:16:47      June 13 02:16:47
```

```
$ MULTINET KERBEROS LIST /SRVTAB
```

```
Server key file: multinet:kerberos.srvtab
```

```
Service      Instance      Realm          Key Version
```

```
-----
```

```
changepw     iris          FLOWERS.COM    1
```

```
rcmd         iris          FLOWERS.COM    1
```

```
$
```

Indicates that CHANGEPW service is configured, as is the RCMD service used by RCP, RLOGIN, and RSHELL.

MULTINET KERBEROS PASSWORD

Changes your Kerberos password.

FORMAT

MULTINET KERBEROS PASSWORD

QUALIFIERS

/INSTANCE="name"

Specifies the instance to change (by default, an empty string).

/REALM=realm

Specifies the Kerberos realm to use. The default is the local realm name specified in the MULTINET:KERBEROS.CONFIGURATION file.

Note! The realm name is case-sensitive.

/USERNAME=login_name

Specifies an alternate *login_name*.

EXAMPLE

```
$ MULTINET KERBEROS PASSWORD
```


MULTINET LPRM

Cancels print jobs, specified by job number, from the SYSS\$PRINT queue. When you issue this command without arguments, the currently active job is cancelled.

FORMAT

MULTINET LPRM *job-ID(s)[,username(s)]*

PARAMETERS

job-ID(s)[,username(s)]

Specifies a comma-separated list of job ID numbers and/or user names. You can only specify job ID numbers of jobs you submitted that originated on your system (unless you are authorized to use /SUPERUSER). Enter a user name to indicate that you want all jobs submitted by the specified user to be removed. If you do not specify /SUPERUSER, you can only specify your user name.

QUALIFIERS

/ALL

Cancels all jobs on the specified printer.

/NODE=remote_print_queue

Specifies the name of a print queue on a remote system.

/QUEUE=queue

Specifies an alternate print queue.

/SUPERUSER

Indicates you have SYSTEM privilege and can delete all jobs in the specified queue.

/USER=user_name

Specifies the user name of the print job to be deleted. To use this qualifier, you must have SYSPRV or OPER privilege.

EXAMPLE

This example invokes LPRM to remove print jobs in the HP_LPD print queue. Job ID numbers 9, 42, and 66 are removed if you submitted them and they originated on your system. In addition, if you are named Lang, all your print jobs are removed from the system. If you are not named Lang, or you did not submit any of the other jobs, the requests are ignored unless you use the /SUPERUSER qualifier.

```
$ MULTINET LPRM /QUEUE=HP_LPD 9,42,66,LANG
```

MULTINET RCP

Transfers file between TCP/IP hosts. Uses the 4.3BSD UNIX "rcp" (remote copy) to copy files between TCP/IP hosts. If the remote host you specify in the input or output file specification is an OpenVMS system running MultiNet, the MultiNet RCP utility automatically negotiates transparent transfer of any OpenVMS file, retaining all RMS attributes.

FORMAT

MULTINET RCP *input_file_spec output_file_spec*

PARAMETERS

input_file_spec

Specifies the name of one or more files to be copied. This parameter may be either a local OpenVMS file specification or a remote file specification of the form:

`hostname::input_file_spec`

If *input_file_spec* is not a full directory and file specification, it is interpreted relative to your login directory on *hostname*. If the directory/file specification on the remote host contains special characters (including mixed-case directory and file names), you should enclose it within double quotation marks.

input_file_spec can be a directory specification if used with the /RECURSIVE qualifier. See the /RECURSIVE qualifier for more details.

You may use wildcards in either the local or remote file specification. For remote file specifications, however, you must use the wildcard characters normally used on the remote system.

output_file_spec

Specifies the name(s) of the output file(s) into which the input file(s) are to be copied. This parameter may be either a local OpenVMS file specification or a remote file specification of the form:

`hostname::output_file_spec`

If *output_file_spec* is not a full directory and file specification, it is interpreted relative to your login directory on *hostname*. If the directory and file specification on the remote host contains special characters (including mixed-case directory and file names), you should enclose it within double quotation marks.

You may use wildcards in either the local or remote file specification. For remote file specifications, however, you must use the wildcard characters normally used on the remote system.

QUALIFIERS

/AUTHENTICATION=KERBEROS

If you specify /AUTHENTICATION=KERBEROS, command authentication is performed using Kerberos; you will not be prompted for authentication information. (KERBEROS is currently the

only value supported by this qualifier.)

/EXACT**/NOEXACT (default)**

Disables the automatic conversion of file names to lowercase. When DCL passes command line parameters and qualifiers to RCP, it converts them to uppercase unless you explicitly enclose them within double quotation marks. Because lowercase file names are preferred by UNIX, and since OpenVMS file names are case-insensitive, RCP converts file names to lowercase. You can use mixed case file names if you enclose them in double quotation marks, and specify them with the /EXACT qualifier.

/LOG=log_spec**/NOLOG (default)**

Specifies whether RCP should display the file specifications and transfer information of each file copied. log_spec can take the values SIZE or TIME (or both if enclosed in parentheses and separated by commas). If you specify only /LOG, /LOG=SIZE is assumed.

When you use the /LOG qualifier, RCP displays the following information for each file copied:

- The names of the input and output files
- The number of blocks copied if you specify /LOG=SIZE
- The data transfer rate (in bytes or kilobytes per second) if you specified /LOG=TIME
- Both the number or blocks and the data transfer rate if you specified /LOG=(SIZE,TIME)

/PASSWORD=password

Specifies the password to use on the remote host which you must specify with the /USERNAME qualifier. If you specify /PASSWORD without a value, RCP prompts for the password (terminal echoing is disabled).

/RECURSIVE**/NORECURSIVE (default)**

Specifies that the directory subtree rooted at the directory named by *input_file_spec* should be copied recursively, that is, the directory and all files and directories below it. If you specify the local file specification with an ellipsis ([...]), the /RECURSIVE qualifier is assumed.

/TRUNCATE_USERNAME**/NOTRUNCATE_USERNAME (default)**

Causes RCP to truncate your OpenVMS user name to be no longer than eight characters. Some RSHELL server implementations, notably UNIX, assume that the remote user name is not longer than eight characters and dies with the error "remuser too long" if it is longer. You can use this qualifier to communicate with those systems.

/USERNAME=username

Specifies the user name to use on the remote host.

/VMS_ATTRIBUTES (default)**/NOVMS_ATTRIBUTES**

Specifies that RCP should attempt to determine if the remote RCP server is another host running MultiNet. If it is, RCP uses a special modification to the "rcp" protocol to transfer OpenVMS file attributes intact. Since this negotiation is compatible with BSD UNIX RCP implementations, it is enabled by default, but may be disabled if compatibility problems arise.

Restrictions

The MultiNet RCP utility does not support third-party copies, so either the input or output file specification may contain remote host information, but not both.

You may use wildcards in either the local or remote file specification. For remote file specifications, however, you must use the wildcard characters normally used on the remote system.

You must specify at least one field in the local file specification. If you do not specify the device or directory, your current default device and directory are used. For a local output specification, RCP fills in any other missing fields (file name, file type, version) with the corresponding field of the input file specification.

RCP fails if a login command procedure displays information. Ensure your OpenVMS login command procedure contains the following lines at the start of the file:

```
$ VERIFY := 'F$VERIFY(0)'  
$ IF F$MODE() .EQS. "OTHER" THEN EXIT
```

You should also add this line to the end of your login command procedure:

```
$ IF VERIFY THEN SET VERIFY
```

For UNIX login scripts (such as .profile), ensure the file does not display any information.

EXAMPLES

This command copies the file JETSON.LOG from your login directory on the host SPROCKETS.COM to your default directory (USERS:[SPACELY]) on the local host.

```
$ RCP SPROCKETS1COM::JETSON.LOG      [ ] /LOG  
%RCP-I-COPIED, SPROCKETS.COM::JETSON.LOG;8  
    copied to USERS:[SPACELY]JETSON.LOG;1 (1 block)
```

This command copies the file LOGIN.COM in your default directory on the local system to the login directory of the user GIGI on the host BIGBOOTE.FLOWERS.COM.

```
$ RCP /USER=GIG2 /PASS=RABBIT LOGIN.COM BIGBOOTE.FLOWERS.COM::
```

In this example, you copy all files in the "tmp" subdirectory of your login directory on the host UNIX.SPROCKETS.COM into your default directory on the local system.

Note! The double quotation marks enclosing "tmp/*" are required to prevent DCL from interpreting the slashes.

```
$ RCP /LOG UNIX.SPROCKETS.COM::"tmp/*" []
%RCP-I-COPIED UNIX.SPROCKETS.COM::tmp/work.order
      copied to USERS:[SPROCKETS]WORK.ORDER;1 (9 blocks)
%RCP-I-COPIED UNIX.SPROCKETS.COM::tmp/judy.note
      copied to USERS:[SPROCKETS]JUDY.NOTE;1 (4 blocks)
%RCP-I-NEWFILES, 2 files created
```

This command copies all directories and files under the "/src" directory tree on UNIX.SPROCKETS.COM. The command creates a comparable directory structure on the local host starting at the current default directory (USERS:[JETSON]), and places the files into this tree. As in the previous example, the double quotation marks enclosing "tmp/*" are required to prevent DCL from interpreting the slashes.

```
$ RCP /RECURSIVE /LOG UNIX.SPROCKETS.COM::"/src" [... ]
%RCP-I-CREATED, created directory USERS:[JETSON.SRC]
%RCP-I-COPIED, UNIX.SPROCKETS.COM::/src/hack.c
      copied to USERS:[JETSON.SRC]HACK.C;1 (20 blocks)
%RCP-I-COPIED UNIX.SPROCKETS.COM::/src/hack.h
      copied to USERS:[JETSON.SRC]HACK.H;1 (2 blocks)
%RCP-I-COPIED created directory USERS:[JETSON.SRC.DATA]
%RCP-I-COPIED, UNIX.SPROCKETS.COM::/src/data/test
      copied to USERS:[JETSON.SRC.DATA]TEST.;1 (100 blocks)
%RCP-I-NEWFILES, 3 files created
```

MULTINET REMIND

Creates reminders to be sent at specified intervals by either mail or broadcast to the recipient's terminal.

FORMAT

MULTINET REMIND

PARAMETERS

After invoking the utility, you are prompted to enter a command. Enter **HELP** to list information about the utility, or enter one of these commands:

Command	Use to...
CREATE	Create new reminders
DELETE <i>nn</i>	Delete a reminder
EXIT	Exit REMIND
LIST	List reminder headers
MODIFY <i>nn</i>	Change an existing reminder
TYPE <i>nn</i>	Display an existing reminder

- *nn* is the reminder number you must supply.

EXAMPLE

In the following example, a question mark is first entered to list possible commands. At each step, a question mark is entered to investigate the possibilities. A reminder is then created and sent.

```
$ REMIND
REMIND>?
CREATE  DELETE  EXIT    HELP    LIST    MODIFY  TYPE
REMIND>CREATE
Time of first reminder? ?
date and time
or one of the following:
FRIDAY      MONDAY      SATURDAY    SUNDAY      THURSDAY
TODAY       TOMORROW    TUESDAY     WEDNESDAY
or one of the following:
APRIL-FOOLS      BASTILLE-DAY      BEETHOVENS-BIRTHDAY
BILBOS-BIRTHDAY  CHRISTMAS          COLUMBUS-DAY
FLAG-DAY         FRODOS-BIRTHDAY   GONDORIAN-NEW-YEAR
GROUND-HOG-DAY   GUY-FAWKES-DAY    HALLOWEEN
INDEPENDENCE-DAY LEAP-DAY           LINCOLNS-BIRTHDAY
```

MAY-DAY MEMORIAL-DAY NEW-YEARS
SAINT-PATRICKS-DAY SHERLOCK-RV-BIRTHDAY VALENTINES-DAY
Time of first reminder? **GROUND-HOG-DAY**
Expiration count? ? Number of times to repeat message
decimal number
Expiration count? 1
How should I send it? ? one of the following:
BOTH MAIL SEND
How should I send it? **MAIL**
Addresses? **HOLMES@FLOWERS.COM**
Subject? **Happy Ground Hog Day!!!**
Text (end with ^Z)
If you see your shadow, consider moving to Santa Cruz.
-Watson
^Z
REMIND> **EXIT**
\$

MULTINET RLOGIN

Connects your terminal to another system on the network. RLOGIN is similar to TELNET, except support for the protocol is not as wide-spread and the protocol automatically authenticates the user instead of requesting a user name and password. Local flow control (instead of remote) is also negotiated dynamically. RLOGIN permits the use of X applications without issuing a SET DISPLAY command.

FORMAT

MULTINET RLOGIN *host_name*

PARAMETERS

host_name

Specifies the remote host to which to connect.

QUALIFIERS

/AUTHENTICATION=KERBEROS

If you specify /AUTHENTICATION=KERBEROS, command authentication is performed using Kerberos; you will not be prompted for authentication information. (KERBEROS is currently the only value supported by this qualifier.)

/BUFFER_SIZE=number

Changes the maximum size of write operations to the terminal. A large write size is more efficient, but a smaller size makes RLOGIN more responsive to output flushing (**Ctrl/O**). The default buffer size is 1024 bytes; the value for number can range from 20 bytes to 1024 bytes. Number is reset to 20 bytes if you specify a value below 20; a value for number above 1024 bytes is reset to 1024.

/DEBUG

Displays any out-of-band control information that arrives during the session.

/EIGHT_BIT

Forces RLOGIN to set the OpenVMS terminal to 8-bit mode for the duration of the session. The default behavior is to use the current setting of the OpenVMS terminal parameter EIGHT_BIT.

/PORT=number

Specifies a non-standard TCP port number to which to connect (the default port is 513).

/TRUNCATE_USERNAME

/NOTRUNCATE_USERNAME (default)

Truncates your VMS user name to a maximum of eight characters. Some RLOGIN server implementations, notably UNIX, assume the remote user name is not longer than eight characters and fail with the error "remuser too long" if it is longer. You can use this qualifier when communicating with such hosts.

/USERNAME=username

Specifies an alternative remote user name. By default, the requested remote user name is the same as your local user name.

EXAMPLE

This example shows an OpenVMS user using RLOGIN to connect to a UNIX system.

```
$ RLOGIN UNIX.FLOWERS.COM
```

```
Last login: Thu Dec 7 22:43:48 from VMS.FLOWERS.COM
```

```
Sun UNIX 4.3 Release 3.5 (UNIX) #1: Fri Apr 7 17:07:00 PDT 2000
```

```
%
```

MULTINET RSHELL

Runs commands on a remote system and displays the command output on your terminal.

FORMAT

MULTINET RSHELL *host_name command_line*

PARAMETERS

host_name

Specifies the remote host on which to execute the command. You can also specify the host name as an IP address.

command_line

Specifies the command line to execute on the remote system. By default, the command line is converted to lowercase. If uppercase characters are required, specify them by enclosing the entire line in double quotations ("*command_line*").

You can specify multiple commands to the OpenVMS RSHELL server by separating them with a backslash-semicolon (;). Ensure the multiple command string does not exceed the DCL limit of 256 bytes for reading command lines.

QUALIFIERS

/ERROR=filename

Specifies the error file name (by default, error output goes to SYS\$ERROR).

/INPUT=filename

Specifies the input file name (by default, SYS\$INPUT). To spawn an RSHELL that does not require input, specify /INPUT=NL: to prevent RSHELL from reading data from your terminal.

/OUTPUT=filename

Specifies the output file name (by default, SYS\$OUTPUT).

/PASSWORD[=password]

Indicates that the REXEC protocol should be used with the specified password instead of the RSHELL protocol. The two protocols are identical except REXEC requires a password, and RSHELL validates on the basis of trusted user names and systems. If you specify /PASSWORD with no password, a password prompt appears with echoing disabled.

/PORT=number

Specifies a non-standard TCP port number to which to connect (by default, port 514 unless you specify /PASSWORD; in that case, port 512 is used).

/TRUNCATE_USERNAME**/NOTRUNCATE_USERNAME (default)**

Truncates your VMS user name to no longer than eight characters. Some RSHELL server implementations, notably UNIX, assume the remote user name is not longer than eight characters and exit with the error "remuser too long" if it is longer. You can use this qualifier to communicate with those systems.

/USERNAME=username

Specifies an alternative remote user name. By default, the remote user name is the same as your local user name.

DESCRIPTION

The MultiNet RSHELL utility uses the 4.3 BSD UNIX rsh (remote shell) protocol to log on, execute a command, and log out. Normally, it authenticates your use of the remote host with its database of trusted hosts and trusted users. However, if you use the /PASSWORD qualifier, the RSHELL utility uses the password you specify and the 4.3 BSD UNIX rexec (remote execution) protocol to authenticate your use of the remote host.

Restrictions

- RSHELL cannot be used to run interactive programs such as editors; use RLOGIN for these applications instead.
- RSHELL permits the use of X Windows applications without the need to issue a SET DISPLAY command.

EXAMPLE

```
$ rshell unix    ls -l
total 216
-rwxr-xr-x 1    smith    212992 Sep 25 07:37 foo
-rw-r--r-- 1    smith      111 Nov 19 22:51 foo.c
$
```

MULTINET RUSERS

Displays information about users logged into local or remote systems. RUSERS can display information about a particular system or, if supported by the network hardware, use broadcasts to display information about all remote systems on directly connected networks. RUSERS uses UDP/IP as its communication protocol.

FORMAT

MULTINET RUSERS [*host-name*]

PARAMETERS

[host-name]

Specifies the name (or network address) of the host from which the remote user information is to be gathered. If you specify the host specified as an asterisk (*), a broadcast RPC gathers information from all directly-connected hosts. If you do not specify a host, a default of * is used.

QUALIFIERS

/ALL

Displays all remote hosts, even those on which there are no users logged in.

/NOALL

Displays only hosts on which there are users logged in (the default).

/FULL

Displays remote users in a longer format, including time of login, idle time, terminal line name, and remote host.

/NOFULL

Displays remote users as a summary line, showing only the system name and user names for that system (the default).

MULTINET SEND

Sends a brief message to another user's terminal.

FORMAT

MULTINET SEND *address [message]*

PARAMETERS

address

Specifies the user name or remote address in the form user@hostname.

Note! Many SMTP implementations do not support the SEND facility that this command uses to send messages.

message

Specifies optional text of the message. If omitted, you are prompted for the message text.

QUALIFIERS

/AND_MAIL

Specifies the message should be both mailed to the user and displayed on the user's terminal.

/OR_MAIL

Specifies the message should be mailed to the user if it cannot be displayed on the user's terminal.

MULTINET TALK

Initiates an interactive conversation with another user on the local host or on any remote host that supports the TALK protocol. Start a conversation by specifying another user's name and host name, if necessary; for example, BILL@FNORD.FOO.COM. End TALKing by pressing **Ctrl/C**. TALK uses the VMS Screen Management (SMG) runtime routines to create a multiwindow display on your terminal through which the conversation takes place. TALK fails if you specify only the person's login name.

FORMAT

MULTINET TALK *user_name*[@*host_name*]

PARAMETERS

user_name

Specifies the remote user name to talk with.

host_name

Specifies the name (or network address) of the host to which a connection should be made. If you do not specify a host name, the local host name is used.

QUALIFIERS

/OLD

Uses the 4.3BSD-compatible TALK protocol. By default, the 4.3BSD-compatible TALK protocol is used. If you are not sure whether to use the new or old TALK, try each. Systems with different system byte-ordering schemes (such as Sun workstations) must use NTALK instead of TALK.

RESTRICTIONS

The restrictions for using TALK include:

You and the person with whom you want to talk need to be on systems with the same byte-ordering scheme (either "Big Endian" or "Little Endian"). While this is not easy to determine, the easiest rule is that if the other person is using a Sun workstation or a terminal connected to one, TALK does not work at their end. Sun users must use the NTALK command. NTALK is provided on the MultiNet software distribution CD-ROM in the [CONTRIBUTED-SOFTWARE.APPLICATIONS.NTALK] directory, and elsewhere as public domain software.

The [CONTRIBUTED-SOFTWARE.APPLICATIONS.NTALK] directory contains documentation describing how to access the file. NTALK is distributed as a UNIX tar file. Use these steps to make it available for use:

1	Copy the NTALK tar archive to a UNIX system.
2	Use tar to retrieve the archived files.

3	<p>Use make to compile the files into binary source. (The make file assumes you have the UNIX cc compiler.)</p> <ul style="list-style-type: none">Both of your terminals must accept broadcasts. Use these commands to enable broadcasts and to suppress mail broadcasts: <pre>\$ SET TERMINAL /BROADCAST \$ SET BROADCAST=NOMAIL</pre> <ul style="list-style-type: none">The terminal type must be listed in the OpenVMS TERMTABLE.TXT database. As shipped with OpenVMS, this database supports all Compaq Computer VT-series terminals. If you have a non-Compaq Computer terminal, check with your system manager.The other person's system must be known to your system. TALK must be able to translate the remote system's IP address into its name. Therefore, your system must be using the Domain Name System (DNS), or have the remote system listed in its host tables.
---	---

USAGE NOTES

Use the following keystrokes during a TALK session:

Press...	To...	Press...	To...
Ctrl/W	Delete the last word typed (left of the cursor)	Ctrl/L	Redraw the screen
Delete	Delete the last character typed	Ctrl/C	Exit to DCL

When someone calls you using TALK, a message similar to the following appears on your terminal:

```
Message from TALK-DAEMON@FLOWERS.COM at 1:53PM-PDT
Connection request by username
[Respond with: TALK username@host]
```

Use this TALK command to answer the remote user's TALK request: `$ TALK username@host`

Once communication is established, you and the other user can type simultaneously, and your output appears in separate windows.

If the user being called has disabled reception of broadcast messages, this message appears:

```
[Your party is refusing messages]
```

EXAMPLE

```
$ TALK HOLMES@FLOWERS.COM
```

MULTINET TELNET

Logs into a remote host from the local host. TELNET uses the standard Internet TELNET protocol to establish a virtual terminal connection between a terminal connected to your VMS system and a remote host.

FORMAT

MULTINET TELNET [*host*]

PARAMETERS

host

Specifies the name or numeric network address of the remote host to which you wish to connect. If you don't specify a host name, TELNET enters a TOPS-20 style interactive mode. If you specify the name of a remote host on the DCL command line, TELNET immediately attempts to connect to the remote host. If you don't specify a remote host, TELNET enters its TOPS-20 style command interpreter and prompts you for TELNET commands.

QUALIFIERS

/ABORT_OUTPUT_CHARACTER=character

Sets the TELNET ABORT-OUTPUT character which, when typed during a TELNET session, sends a TELNET ABORT OUTPUT sequence to the remote host. Specify control characters with a caret (^) followed by a letter. By default, there is no ABORT OUTPUT character; specifying this qualifier without a value sets the character to ^O (a caret followed by uppercase O, to represent **Ctrl/O**).

/ARE_YOU_THERE_CHARACTER=character

Sets the TELNET ARE-YOU-THERE character which, when typed during a TELNET session, sends a TELNET ARE YOU THERE sequence to the remote host. By default, there is no ARE-YOU-THERE character; specifying that qualifier without a value sets the character to ^T (a caret followed by uppercase T, to represent **Ctrl/T**).

/AUTHENTICATION=KERBEROS

Uses the Kerberos authentication system.

/AUTOFLUSH

Activates the AUTOFLUSH feature. When used with the /ABORT_OUTPUT_CHARACTER, /BREAK_CHARACTER, and /INTERRUPT_PROCESS_CHARACTER qualifiers, the /AUTOFLUSH qualifier causes TELNET to flush any data which may be in the network buffers when the ABORT-OUTPUT, INTERRUPT_PROCESS, or BREAK character is used. Data is flushed by sending a TIMING-MARK command to the TELNET server and discarding all data until one is received in response.

/BREAK_CHARACTER=character

Sets the TELNET BREAK character which, when typed during a TELNET session, sends a

TELNET BREAK sequence to the remote host. By default there is no BREAK character; specifying this qualifier without a value sets the character to ^C (a caret followed by uppercase C, to represent `Ctrl/C`).

/BUFFER_SIZE=number

Changes the maximum size of terminal write operations to the specified *number* of bytes. A large write size is more efficient, but a smaller size makes TELNET more responsive.

The default buffer size is 512 bytes. The value for *number* can range from 20 to 1024 bytes. If you specify a value below 20, the buffer size is reset to 20. If you specify a value above 1024, it is reset to 1024.

/CREATE_NTY[(options)]

Performs the same function as the CREATE-NTY command (available in command mode once a connection has been made). When specified without options, /CREATE_NTY causes TELNET to make a temporary connection to the specified host, attach this connection to an NTY device, and exit immediately. You can then run another application, such as KERMIT or SET HOST/DTE through this pseudo-terminal device. The TELNET_NTY logical name is defined to be the NTY device created. Use it as you would any other terminal device. When you are finished with the terminal, use the DEALLOCATE command to dismantle the connection and associated NTY device control blocks. Alternatively, the connection will be dismantled when you log out.

```
$ TELNET /CREATE_NTY[( [PERMANENT] -
    [,NAME=logical_name] -
    [,TABLE=logical_name_table] -
    [,MODE={EXECUTIVE|SUPERVISOR}] -
    [/PORT=target-TCP-port] -
host
```

The *options* contain a comma-separated list beginning with:

PERMANENT	Specifies that the NTY device will persist after you close the TELNET connection. To delete the permanent NTY device, use the MULTINET TELNET /DELETE_NTY=logical_name command.
and continuing with any of the following:	
NAME= <i>logical_name</i>	Specifies the NTY device's logical name. The default logical name is TELNET_NTY.
TABLE= <i>logical_name_table</i>	Specifies the logical name table to which the new NTY device name is added. The default logical name table is LNM\$PROCESS_PROCESS.
MODE= <i>access_mode</i>	Specifies the logical name's access mode. <i>access_mode</i> is either SUPERVISOR (the default) or EXECUTIVE.

Privileged users can use /CREATE_NTY options to establish permanent NTY devices. In this case, the NTY device is created but no connection is made to the specified host until the first I/O operation.

Use this qualifier only with permanent NTY devices.

/DELETE_NTY=*logical_name*

Deletes a permanent NTY device named by *logical_name*. Create permanent NTY devices with the MULTINET TELNET /CREATE_NTY command.

/DEBUG

/NODEBUG (default)

Sets the TELNET debug flag. When you specify /DEBUG, TELNET prints all option negotiations made with the remote host.

/ERASE_CHARACTER_CHARACTER=*character*

Sets the TELNET ERASE-CHARACTER character which, when typed during a TELNET session, sends a TELNET ERASE CHARACTER sequence to the remote host. By default, there is no ERASE-CHARACTER character. Specifying this character without a value sets this character to ^? (a caret followed by a question mark, to represent **Delete**).

/ERASE_LINE_CHARACTER=*character*

Sets the TELNET ERASE-LINE character which, when typed during a TELNET session, sends a TELNET ERASE LINE sequence to the remote host. By default, there is no ERASE LINE character; specifying this qualifier without a value sets the character to ^U (a caret followed by uppercase U, representing **Ctrl/U**).

/ESCAPE_CHARACTER=*character*

Sets the TELNET ESCAPE character. When you type the TELNET ESCAPE character during a TELNET session, communication with the remote host temporarily stops, and TELNET interprets the next character you type as a TELNET command. The ESCAPE character defaults to ^^ (two consecutive carets, representing **Ctrl/^**).

After you type the TELNET ESCAPE character, the next character you type is interpreted according to the following list:

Character	Action
?	Displays information about TELNET escape commands.
A	Sends an INTERRUPT PROCESS command to the remote host.
B	Sends a BREAK command to the remote host.
C	Closes the connection to the remote host.
O	Sends an ABORT OUTPUT command to the remote host.

Character	Action
P	Spawns a new DCL process (or attaches to a parent process).
S	Displays the status of the TELNET connection.
T	Sends an ARE YOU THERE (AYT) command. On a MultiNet server, this command is mapped to Ctrl/T .
Q	Quits TELNET.
X	Enters extended TELNET command mode.

Type the ESCAPE character twice to send it to the remote host.

/INTERRUPT_PROCESS_CHARACTER=character

Sets the TELNET INTERRUPT-PROCESS character which, when typed during a TELNET session, sends an INTERRUPT PROCESS sequence to the remote host. By default, there is no INTERRUPT PROCESS character; specifying this qualifier without a value sets the character to ^C (a caret followed by uppercase C, representing **Ctrl/C**).

/LOCAL_FLOW_CONTROL

/NOLOCAL_FLOW_CONTROL

Specifies that **Ctrl/Q** and **Ctrl/S** should be treated by the local terminal driver as XON and XOFF, instead of being passed down the network connection for processing by the remote terminal driver. Use of this qualifier makes XOFF more responsive, which helps prevent data loss; however, the remote system will never see any **Ctrl/S** character.

The default flow control setting depends on the setting of the VMS terminal characteristic TT\$_TTSYNC (set by the DCL command SET TERMINAL /TTSYNC or by many full-screen editors). Specify **/LOCAL_FLOW_CONTROL** to force TELNET into local flow control mode. Specify **/NOLOCAL_FLOW_CONTROL** to force TELNET into remote flow control mode.

/LOG_FILE=[file-spec]

/NOLOG_FILE (default)

Specifies a file in which to log a transcript of the TELNET session. Everything received by the local system from the remote system is recorded in this file. If you specify the **/LOG_FILE** qualifier without a value, the default file specification TELNET.LOG is used. The log file is created in the directory from which TELNET is run. **/LOG_FILE** is not supported in 3270 or 5250 mode.

/PORT=port-spec

Specifies the port to which a connection is to be made. If you do not specify this qualifier, the standard TELNET port for the specified protocol is used. For the TCP/IP protocol, use a port number or a port defined in MULTINET:HOSTS. service file.

When connecting via TCP/IP to a port other than the default TELNET port (23), full VMS command line editing is available on command input.

/PRINT_ESCAPE_CHARACTER

Displays the ESCAPE character used to access TELNET command mode. If you use this qualifier, the escape character is displayed when a connection occurs:

Escape character is '^'^'

You can also use the logical name MULTINET_TELNET_PRINT_ESCAPE_CHARACTER to set this feature.

/PROTOCOL=protocol-spec

Specifies the protocol to be used in making the connection to the remote system. The protocol specification can be either TCP or IP (TCP is the default).

/TCP

Used as an abbreviation for /PROTOCOL=TCP.

/TERMINAL_TYPE

Specifies the terminal type to be negotiated with the remote TELNET server. This qualifier has the same function as the TERMINAL-TYPE command.

/TN3270=AUTOMATIC (default)**FORCE****/NOTN3270**

Allows the negotiation of IBM 3270 terminal emulation mode. AUTOMATIC (the default) causes TELNET to automatically negotiate IBM 3270 emulation mode with the remote host. TELNET enters 3270 mode only if the remote host supports it.

Use FORCE to force TELNET into IBM 3270 emulation mode when communicating with a system that supports 3270 mode, but cannot negotiate it automatically. (IBM mainframes running ACCESS/MVS have this restriction.) Use /NOTN3270 to disable IBM 3270 emulation mode entirely.

/TN5250=AUTOMATIC (default)**FORCE****/NOTN5250**

Allows the negotiation of IBM 5250 terminal emulation mode. Use AUTOMATIC (the default) to cause TELNET to automatically negotiate IBM 5250 emulation mode with the remote host. TELNET enters 5250 mode only if the remote host supports it. FORCE is used to force TELNET into IBM 5250 emulation mode when communicating with a system that supports 5250 mode, but cannot negotiate it automatically. IBM MVS does not support 5250. Use /NOTN5250 to disable IBM 5250 emulation mode entirely.

/UNIX**/NOUNIX (default)**

Uses the 4.3BSD UNIX end-of-line specification, <CR><NL>, instead of the standard end-of-line specification, <CR><LF>. This qualifier is useful when using TELNET to connect to 4.3BSD UNIX systems.

/VERSION

Displays version information about the TELNET utility. If you use this qualifier, all other parameters and qualifiers are ignored and a TELNET session is not started.

Note! To specify a control character for the value of character in any of the preceding qualifiers, type it as a ^ (caret) followed by the appropriate character, all enclosed within double quotes.

EXAMPLES

This command creates a permanent NTY device pointing at port 9100 on WHORFIN.FLOWERS.COM, and creates the logical name WHORFINDEVICE (in the system logical name table in executive mode) that translates to the NTY device name.

```
1 $ MULTINET TEL1ET FLOWERS.COM
2 $ MULTINET TELNET SALES.FLOWERS.COM /LOG_FILE=SALES.LOG
3 $ MULTINET TELNET LOCALHOST /PORT=SMTP
4 $ MULTINET TELNET /ABORT_OUTPUT_CHARACTER="^A"
5 $ MULTINET TEL5ET /PORT=9100 /CREATE_nty=PERMANENT,
   _$NAME=WHORFINDEVICE, TABLE=SYSTEM, MODE=EXECUTIVE -
   _$ WHORFIN.FLOWERS.COM
```

MULTINET TFTP

Transfers files between TCP/IP hosts. The TFTP utility uses the Internet standard Trivial File Transfer Protocol to transfer files between Internet hosts. TFTP uses the User Datagram Protocol (UDP), and performs no user authentication.

FORMAT

TFTP [*host* [*port*]]

PARAMETERS

host

Specifies the name or numeric address of the remote host to which you want to connect.

port

Specifies the UDP port number on the server to which you want to connect. If you don't specify the port number, the standard TFTP UDP server port number (69) is used.

EXAMPLE

This example shows how to use TFTP to connect to the host FLOWERS.COM.

```
$ TFTP FLOWERS.COM  
tftp>
```

MULTINET WHOIS

Displays information about users registered with the Internet Network Information Center (InterNIC). The default WHOIS server is RS.INTERNIC.NET.

FORMAT

MULTINET WHOIS *name*

PARAMETERS

name

Specifies the name or *handle* of the registered user about whom you want to retrieve information.

For more information and help from the InterNic type WHOIS HELP from the DCL prompt.

QUALIFIERS

/HOST=hostname

Specifies the remote host to which to connect. The default is RS.INTERNIC.NET, but can be changed by a system manager. The connection is done to the NICNAME port.

/OUTPUT=filespec

Specifies an output file in which to store WHOIS output.

/PORT= port number

Specifies the number of a non-standard port.

EXAMPLE

This example shows how to display information about the user "Smith" from the InterNIC database.

```
$ WHOIS SMITH
```

```
SMITH, J.R.    smith@abc.com
ABC, Incorporated
101 Elm Street
Surf City, CA 95060
(408) 555-1212
Record last updated on 1-Jun-00.
```

The InterNIC Registration Services Host ONLY contains Internet Information Networks, ASN's, Domains, and POC's).

Appendix B

FTP Command Reference

The MultiNet FTP utility uses the Internet-standard FTP (File Transfer Protocol) to transfer files between the local host and a remote host. This appendix lists the commands you can use during an FTP session.

FTP Command Summary

The following table lists the MultiNet FTP commands:

Table B-1 FTP Command Summary

Command	Description
ACCOUNT	Sends an account name to the remote FTP server.
AGET	Appends a remote file to a file on the local host.
APPEND GET	Appends <i>remote_file</i> from the remote host to <i>local_file</i> on the local host.
APPEND PUT	Appends <i>local_file</i> on the local host to <i>remote_file</i> on the remote host.
APPEND RECEIVE	Appends <i>remote_file</i> from the remote host to <i>local_file</i> on the local host.
APPEND SEND	Appends <i>local_file</i> on the local host to <i>remote_file</i> on the remote host.
APUT	Appends <i>local_file</i> on the local host to <i>remote_file</i> on the remote host.
ASCII	Sets the transfer type to ASCII for transferring text files.

Table B-1 FTP Command Summary (Continued)

Command	Description
ATTACH	Detaches the terminal from the calling process and reattaches it to another process.
BELL	Turns on, off, or toggles the sounding of a bell when a file transfer completes.
BINARY	Sets the transfer type for transferring binary files.
BLOCK	Reads files of TYPE I, STRUCTURE FILE using block I/O.
BYE	Closes the current FTP connection, but remains in the FTP command interpreter.
BYTE	Sets the transfer byte size to size.
CD	Changes the current working directory on the remote host to dir.
CDUP	Changes the current working directory on the remote host by moving up one level in the directory system.
CLOSE	Closes the current FTP connection, but remains in the FTP command interpreter.
CONFIRM	Turns on, off, or toggles (the default) interactive confirmation of each command in a MULTIPLE command.
CONNECT	Establishes a connection to the FTP server on host.
CPATH	Changes the current working directory on the remote host to dir.
CREATE-DIRECTORY	Creates the directory dir on the remote host.
CWD	Changes the current working directory on the remote host to dir.
DELETE	Deletes a file on the remote host.
DIRECTORY	Obtains an annotated listing of the files on the remote host.
DISCONNECT	Closes the current FTP connection without waiting for a confirming response from the remote host, but remains in the FTP command interpreter.
EXIT	Closes the current FTP connection and exits FTP.
EXIT-ON-ERROR	Turns on, off, or toggles (the default) whether or not FTP automatically exits when an error occurs.

Table B-1 FTP Command Summary (Continued)

Command	Description
GET	Copies <i>remote_file</i> from the remote host to <i>local_file</i> on the local host.
HASH	Turns on, off, or toggles (the default) the display of hash marks (#) for each data buffer transferred.
HELP	Displays FTP help information.
LCD	Changes the current working directory on the local host to <i>dir</i> .
LDIR	Displays the contents of your local working directory. LDIR is the same as LOCAL-DIRECTORY.
LIST	Displays automatic login information for <i>host</i> .
LOCAL-CD	Changes the current working directory on the local host to <i>dir</i> .
LOCAL-DIRECTORY	Displays the contents of your local working directory.
LOCAL-PWD	Displays the current working directory on the local host.
LOGIN	Identifies you to a remote FTP server.
LPWD	Displays the current working directory on the local host.
LS	Displays a names-only listing of files on the remote host.
MDELETE	Deletes multiple files on the remote host.
MGET	Copies multiple files from the remote host to the local host.
MKDIR	Creates the directory <i>dir</i> on the remote host.
MODE	Sets the transfer mode to COMPRESSED or STREAM (the default).
MPUT	Copies multiple files from the local host to the remote host.
MULTIPLE DELETE	Deletes multiple files on the remote host.
MULTIPLE GET	Copies multiple files from the remote host to the local host.
MULTIPLE PUT	Copies multiple files from the local host to the remote host. MULTIPLE PUT is a synonym for MULTIPLE SEND. See MULTIPLE SEND for more information.
MULTIPLE RECEIVE	Copies multiple files from the remote host to the local host.
MULTIPLE SEND	Copies multiple files from the local host to the remote host.

Table B-1 FTP Command Summary (Continued)

Command	Description
OPEN	Establishes a connection to a host system.
PASSIVE	Enables or disables "passive" mode for file transfers with FTP servers on the opposite side of "firewall" gateways.
PASSWORD	Sends a password to the remote FTP server explicitly, which normally happens automatically during login.
PORT	Specifies a TCP port number to use for the FTP control connection.
PROMPT-FOR-MISSING-ARGUMENTS	Turns on, off, or toggles (the default) whether or not FTP automatically prompts for missing command arguments.
PROMPT-ON-CONNECT	Turns on, off, or toggles (the default) whether or not FTP automatically prompts for a user name and password after making a connection.
PUSH	Starts and attaches a DCL subprocess.
PUT	Copies <i>local_file</i> on the local host to <i>remote_file</i> on the remote host.
PWD	Displays the current working directory on the remote host.
QUIT	Closes the current FTP connection and exits FTP.
QUOTE	Sends a string to the FTP server verbatim.
RECEIVE	Copies remote-file from the remote host to local-file on the local host.
RECORD-SIZE	Sets or displays the record size for IMAGE mode transfers.
REMOTE-HELP	Displays information about commands available on the FTP server.
REMOVE-DIRECTORY	Deletes a directory on the remote host. REMOVE-DIRECTORY is the same as RMDIR.
RENAME	Renames files on the remote host.
RETAIN	Turns on, off, or toggles (the default) the retention of OpenVMS version numbers in file transfers.
RM	Deletes a file on the remote host.
RMDIR	Deletes a directory on the remote host.

Table B-1 FTP Command Summary (Continued)

Command	Description
SEND	Copies <i>local_file</i> on the local host to <i>remote_file</i> on the remote host.
SET	Sets automatic login information for host.
SHOW-DIRECTORY	Displays the current working directory on the remote host. SHOW DIRECTORY is the same as PWD.
SITE	Specifies commands that are interpreted by the MultiNet FTP server for use on the server host.
SPAWN	Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH.
STATISTICS	Turns on, off, or toggles (the default) STATISTICS mode.
STATUS	Displays the status of the FTP server.
STREAM	Turns on, off, or toggles (the default) the creation of binary output files as Stream_LF files.
STRUCTURE	Sets the transfer structure to <i>structure</i> .
TAKE	Interprets FTP commands in a file.
TENEX	Changes the byte size for transferring binary files to or from a TOPS-20 system.
TYPE	Sets the transfer type to <i>type</i> .
USER	Identifies you to the remote FTP server.
VERBOSE	Turns on, off, or toggles (the default) VERBOSE mode.
VERSION	Prints information about the FTP program_version.

ACCOUNT

Sends an account name to the remote FTP server. Use this command when connecting to hosts that require account specifications in addition to a user name.

FORMAT

ACCOUNT *account*

PARAMETERS

account

Specifies the name of the account to be sent to the remote FTP server.

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

```
FLOWERS.COM> account sales  
<Account "sales" accepted  
FLOWERS.COM>
```

AGET

Appends a remote file to a file on the local host. AGET is a synonym for APPEND GET. See APPEND GET for more information.

FORMAT

AGET *remote_file* [*local_file*]

APPEND GET

Appends remote_file from the remote host to local_file on the local host. APPEND uses the current settings for type, mode, and structure during file transfers. APPEND GET is the same as AGET and APPEND RECEIVE.

FORMAT

APPEND GET *remote-file* [*local-file*]

PARAMETERS

remote_file

Specifies the name of the file on the remote host from which to copy.

local_file

Specifies the name of a file on the local host to which the file is to be appended.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the GET command.
- You cannot use the APPEND GET command in STRUCTURE VMS mode. If you try to do this, FTP toggles temporarily into STRUCTURE FILE mode for the transfer.

EXAMPLE

This example shows how to append a remote file to a file on the local host.

```
FLOWERS.COM> append get login.com
To local file: RETURN
<ASCII retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
<Transfer completed.  2498 (8) bytes transferred.
FLOWERS.COM>
```


APPEND PUT

Appends *local_file* on the local host to *remote_file* on the remote host. APPEND PUT is a synonym for APPEND SEND. See APPEND SEND for more information.

FORMAT

APPEND PUT *local_file remote_file*

APPEND RECEIVE

Appends *remote_file* from the remote host to *local_file* on the local host. APPEND RECEIVE is a synonym for APPEND GET. See APPEND GET for more information.

FORMAT

APPEND RECEIVE *remote_file* [*local_file*]

APPEND SEND

Appends *local_file* on the local host to *remote_file* on the remote host. APPEND SEND uses the current settings for type, mode, and structure during file transfers. APPEND SEND is the same as APUT and APPEND PUT.

FORMAT

APPEND SEND *local_file remote_file*

PARAMETERS

local_file

Specifies the name of the file on the local host to be copied.

remote_file

Specifies the destination file name on the remote host.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the APPEND SEND command.
- The MultiNet FTP Server cannot APPEND to a file in STRUCTURE VMS mode.

EXAMPLE

This example shows how to append the LOGIN.COM file to the remote file FOO.COM.

```
FLOWERS.COM>append send login.com foo.com
<ASCII Store of ST_ROOT:[TMP]FOO.COM;12 started.
<Transfer completed. 2498 (8) bytes transferred.
FLOWERS.COM>
```

APUT

Appends *local_file* on the local host to *remote_file* on the remote host. APUT is a synonym for APPEND PUT and APPEND SEND. See APPEND SEND for more information.

FORMAT

APUT *local_file remote_file*

ASCII

Sets the transfer type to ASCII for transferring text files. ASCII is a synonym for TYPE ASCII. See TYPE for more information.

FORMAT

ASCII

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the SPAWN SHOW PROCESS /SUBPROCESSES command to list the names of subprocesses. Use the DCL LOGOUT command to return to the original process. If the MULTINET_DISABLE_SPAWN logical is enabled, ATTACH does not work.

FORMAT

ATTACH *process-name*

PARAMETERS

process_name

Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

BELL

Turns on, off, or toggles the sounding of a bell when a file transfer completes.

FORMAT

BELL *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to toggle the bell feature.

```
FTP>bell  
[Bell will now ring when operations complete]  
FTP>
```

BINARY

Sets the transfer type for transferring binary files. BINARY is a synonym for TYPE IMAGE. See TYPE for more information.

FORMAT

BINARY

BLOCK

Reads files of TYPE I, STRUCTURE FILE using block I/O.

FORMAT

BLOCK

Restrictions

Use this command only when connected to a remote host.

BYE

Closes the current FTP connection, but remains in the FTP command interpreter.

FORMAT

BYE

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

This example shows how to disconnect an FTP connection.

```
FLOWERS.COM>bye  
<QUIT command received. Goodbye.  
FTP
```

BYTE

Sets the transfer byte size to size.

FORMAT

BYTE *size*

PARAMETERS

size

Specifies the size to which to set the transfer byte size. The only permitted value is 8 bits.

EXAMPLE

This example shows how to set the transfer byte size to 8 bits.

```
FLOWERS.COM>byte
Type: Logical-Byte (Byte Size 8), Structure: VMS, Mode: Stream
FLOWERS.COM>
```

CD

Changes the current working directory on the remote host to *dir*. CD is the same as CPATH and CWD.

FORMAT

CD *dir*

PARAMETERS

dir

Specifies the name of the directory to use as the current working directory.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the CD command.

EXAMPLE

This example shows how to change the default directory on the remote host to USERS:[ANONYMOUS].

```
FLOWERS.COM>cd [anonymous]  
<Connected to USERS:[ANONYMOUS].  
FLOWERS.COM>
```

CDUP

Changes the current working directory on the remote host by moving up one level in the directory system.

FORMAT

CDUP

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the CDUP command.

EXAMPLE

This example shows how to move up one directory on the remote host.

```
FLOWERS.COM>cdup  
<Connected to USERS:[000000].  
FLOWERS.COM>
```

CLOSE

Closes the current FTP connection, but remains in the FTP command interpreter. CLOSE is a synonym for BYE. See BYE for more information.

FORMAT

CLOSE

CONFIRM

Turns on, off, or toggles (the default) interactive confirmation of each command in a MULTIPLE command.

FORMAT

CONFIRM *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to enable CONFIRM mode and use it with MGET to prompt for each file name.

```
FLOWERS.COM>confirm
[You will be asked to confirm each transaction in a multiple transaction]
FLOWERS.COM>mget *.com
<List started.
<Transfer completed.
GET copy.com? [YES] n
GET login.com? [YES] y
<VMS retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
<Transfer completed. 2498 (8) bytes transferred.
FLOWERS.COM>
```

CONNECT

Establishes a connection to the FTP server on *host*. CONNECT is the same as OPEN.

FORMAT

CONNECT *host*

PARAMETERS

host

Specifies the name of the host to which to establish a connection. *host* is specified as either a symbolic host name or as a dotted Internet address.

Restrictions

Do not use this command when connected to a remote host.

EXAMPLE

This example shows how to connect to the FLOWERS.COM host.

```
FTP>connect flowers.com
Connection opened (Assuming 8-bit connections)
<FLOWERS.COM MultiNet FTP Server Process
<4.0 (nnn) at Fri 7-Apr-2000 7:42am-PST
FLOWERS.COM>
```


CPATH

Changes the current working directory on the remote host to *dir*. CPATH is a synonym for CD. See CD for more information.

FORMAT

CPATH *dir*

CREATE-DIRECTORY

Creates the directory *dir* on the remote host. CREATE DIRECTORY is the same as MKDIR.

FORMAT

CREATE-DIRECTORY *dir*

PARAMETERS

dir

Specifies the name of the directory to create.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the CREATE-DIRECTORY command.

EXAMPLE

This example shows how to create the subdirectory "test".

```
FLOWERS.COM>create-dir test  
<"USERS:[HOLMES.TEST]" Directory created  
FLOWERS.COM>
```

CWD

Changes the current working directory on the remote host to *dir*. CWD is a synonym for CD. See CD for more information.

FORMAT

CWD *dir*

DELETE

Deletes a file on the remote host. DELETE is the same as RM.

FORMAT

DELETE *file*

PARAMETERS

file

Specifies the name of the file to delete.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the DELETE command.

EXAMPLE

This example shows how to delete the file FOO.BAR from the remote host.

```
FLOWERS.COM>del foo.bar  
<File deleted ok, file USERS:[HOLMES]FOO.BAR;1.  
FLOWERS.COM>
```

DIRECTORY

Obtains an annotated listing of the files on the remote host.

FORMAT

DIRECTORY [*file_spec*] [*output_file*]

PARAMETERS

file_spec

Specifies the file specification to use in the directory lookup on the remote host. If you do not specify *file_spec*, the current working directory on the remote host is used. Any wildcards you specify are interpreted in the context of the remote host operating system.

output_file

Specifies the name of the file to which to write the directory listing. If you do not specify *output_file*, the list is directed to SYS\$OUTPUT:.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the **DIRECTORY** command.

EXAMPLE

This example shows how to retrieve list of files that match the wildcard "*.COM".

```
FLOWERS.COM>dir *.com
<List started.
USERS: [HOLMES]
COPY.COM;4      2      1-APR-2000 08:49 [HOLMES] (RWD,RWD,R,R)
LOGIN.COM;1     5      1-APR-2000 01:25 [HOLMES] (RWD,RWD,R,R)
Total of 7 blocks in 2 files.
<Transfer completed.
FLOWERS.COM>
```

DISCONNECT

Closes the current FTP connection without waiting for a confirming response from the remote host, but remains in the FTP command interpreter.

FORMAT

DISCONNECT

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

```
FLOWERS.COM>disc  
FTP>
```

EXIT

Closes the current FTP connection and exits FTP. QUIT is the same as EXIT.

FORMAT

EXIT

EXAMPLE

```
FLOWERS.COM>exit  
<QUIT command received. Goodbye.  
$
```

EXIT-ON-ERROR

Turns on, off, or toggles (the default) whether or not FTP automatically exits when an error occurs.

If EXIT-ON-ERROR is enabled, FTP automatically exits if an error occurs. After exiting, the DCL symbol \$STATUS contains the status code of the last error to occur. If the last error was reported by the FTP server, it contains the value %X1000002C + (%X10000 * *ftp_error_code*).

FORMAT

EXIT-ON-ERROR *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to use EXIT-ON-ERROR to automatically exit when an error occurs. Here the error was not an FTP error.

```
FTP>exit-on-error
[Will exit when an error occurs]
FTP>connect 1.2.3.4
1.2.3.4: %MULTINET-F-ETIMEDOUT, Connection timed out
$ sho symbol $status
$STATUS == "%X100081E4"
```

This example shows how EXIT-ON-ERROR exits automatically when an error occurs. Here the FTP Server responded as follows to the command **user unknown password**:

```
FTP>exit-on-error
[Will exit when an error occurs]
FTP>connect somehost
Connection opened (Assuming 8-bit connections)
<Somehost MultiNet FTP Server Process V4.3(15) at Thu 4-May-00 2:37PM-PDT
SOMEHOST>user unknown password
<%SYSTEM-F-INVLOGIN, login information invalid at remote node
$ show symbol $status
$STATUS == "%X1212002C"
$ write sys$output ($status-%X1000002C)/%X10000
530

530 %SYSTEM-F-INVLOGIN, login information invalid at remote node
```


GET

Copies *remote_file* from the remote host to *local_file* on the local host. The current settings for type, mode, and structure are used during file transfers. GET is the same as RECEIVE.

FORMAT

GET remote-file [*local-file*]

PARAMETERS

remote-file

Specifies the name of the file on the remote host.

local-file

Specifies the name of the file on the local host.

QUALIFIERS

/FDL

Obtains a file previously saved with the PUT /FDL command. When you create a file with the PUT /FDL qualifier, a file description language (FDL) file is created at the same time as the original file. The output file is converted to raw block format. When you retrieve a file with GET /FDL, the original format is restored using the attributes stored in the FDL file. If you don't use the /FDL qualifier with the GET command, the new raw block format is retained. In any case, the FDL file is retained and must be deleted independently. The /FDL qualifier provides compatibility with DEC TCP/IP Services for OpenVMS (formerly UCX). The FDL file has the same name except the string FDL is appended to the end.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the GET command.

EXAMPLE

This example shows how to transfer a file to the local host.

```
FLOWERS.COM>get login.com
To local file: RETURN
<VMS retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
<Transfer completed. 2498 (8) bytes transferred.
FLOWERS.COM>
```

HASH

Turns on, off, or toggles (the default) the display of hash marks (#) for each data buffer transferred.

FORMAT

HASH *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to display hash marks during a GET file transfer.

```
FLOWERS.COM>hash
[Hash marks will be printed during transfers]
FLOWERS.COM>get login.com login.com
<VMS retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
###
Transfer completed. 2498 (8) bytes transferred.
```

HELP

Displays FTP help information. Type `HELP ?` to see a list of HELP topics. Type `HELP` without an argument to display general help information.

FORMAT

HELP *[command]*

PARAMETERS

command

Specifies the name of the command about which you want help.

EXAMPLE

FTP>**help**

The `HELP` command prints on-line help for the FTP user program. The argument to `HELP` selects the particular FTP command about which help is desired. In addition to the FTP commands, several control characters can be typed while file transfers are in progress:

Control-A shows the progress of a data transfer.

Control-G aborts the file transfer and returns to FTP command level.

Control-P spawns a new command interpreter.

FTP>

LCD

Changes the current working directory on the local host to *dir*. LCD is a synonym for LOCAL-CD. See LOCAL-CD for more information.

FORMAT

LCD *dir*

LDIR

Displays the contents of your local working directory. LDIR is the same as LOCAL-DIRECTORY.

FORMAT

LDIR

EXAMPLE

```
FTP>ldir *.com
USERS:[FLOWERS.DOC.V32]
DOC.COM;2      1    1-APR-2000 01:36 FLOWERS_FILES (RWED,RWED,,)
LOGIN.COM;3    5    1-APR-2000 19:07 FLOWERS_FILES (RWED,RWED,,)
LOGIN.COM;2    5    1-APR-2000 19:04 FLOWERS_FILES (RWED,RWED,,)
LOGIN.COM;1    5    1-APR-2000 18:49 FLOWERS_FILES (RWED,RWED,,)
Total of 16 blocks in 4 files.
FTP>
```

LIST

Displays automatic login information for *host*. See the SET command for information about setting automatic login information for a host.

FORMAT

LIST [*host*]

PARAMETERS

host

Specifies the host whose automatic login information you are trying to display. If you do not specify host, LIST displays automatic login information for all hosts for which login information has been set.

Restrictions

Do not use this command when connected to a remote host.

EXAMPLE

This example shows how to set and list information for the DS.INTERNIC.NET host.

```
FTP>set ds.internic.net /user=anonymous /pass=guest
FTP>list
DS.INTERNIC.NET
    User: anonymous
    Password: guest
FTP>
```

LOCAL-CD

Changes the current working directory on the local host to *dir*. LOCAL-CD is the same as LCD.

FORMAT

LOCAL-CD *dir*

PARAMETERS

dir

Specifies the name of the directory to which to change the current working directory.

EXAMPLE

```
FTP>lcd [-]  
Connected to USERS:[FLOWERS.DOC].  
FTP>
```

LOCAL-DIRECTORY

Displays the contents of your local working directory. LOCAL-DIRECTORY is a synonym for LDIR. See LDIR for more information.

FORMAT

LOCAL-DIRECTORY

LOCAL-PWD

Displays the current working directory on the local host. LOCAL-PWD is a synonym for LPWD.

FORMAT

LOCAL-PWD

LOGIN

Identifies you to a remote FTP server. LOGIN is the same as USER.

FORMAT

LOGIN **user** [*password*]

PARAMETERS

user

Specifies your user name on the remote server.

password

Specifies your password on the remote server. If you do not specify password and the remote site requires one, you are prompted for it. In either case, the password is not echoed.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts do not allow you to use LOGIN once you have already logged in.

EXAMPLE

This example shows how to connect to a remote host and log in.

```
$ ftp irisd.com
Connection opened (Assuming 8-bit connections)
<IRISD.COM MultiNet FTP Server Process 4.0(nn) at Fri 7-Apr-2000 7:42amPST
IRISD.COM>login HOLMES password
<User HOLMES logged into U1:[HOLMES] at Fri 7-Apr-2000, 19:13, job 433.
IRISD.COM>
```

LPWD

Displays the current working directory on the local host. LPWD is the same as LOCAL-PWD.

FORMAT

LPWD

EXAMPLE

```
FTP>lpwd
Connected to USERS:[FLOWERS.DOC].
FTP>
```

LS

Displays a names-only listing of files on the remote host. You can use wildcard specifications.

FORMAT

LS [*file_spec*] [*output_file*]

PARAMETERS

file_spec

Specifies the file specification to use in the directory lookup on the remote host. If you do not specify *file_spec*, the current working directory on the remote host is used. Any wildcards used are interpreted in the context of the remote host operating system.

output_file

Specifies the name of the file to which to write the directory listing. If *output_file* is not specified, the list is directed to SYS\$OUTPUT:.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the LS command.

EXAMPLE

This example shows how to retrieve the directory listing of the files matching the wildcard character *.

```
FLOWERS.COM>ls *.
<List started.
$mailinterface.
mymail.
todo.
<Transfer completed.
FLOWERS.COM>
```

MDELETE

Deletes multiple files on the remote host. MDELETE is a synonym for MULTIPLE DELETE. See MULTIPLE DELETE for more information.

FORMAT

MDELETE *files*

MGET

Copies multiple files from the remote host to the local host. MGET is a synonym for MULTIPLE GET. See MULTIPLE GET for more information.

FORMAT

MGET *files*

MKDIR

Creates the directory *dir* on the remote host. MKDIR is a synonym for CREATE-DIRECTORY. See CREATE-DIRECTORY for more information.

FORMAT

MKDIR *dir*

MODE

Sets the transfer mode to COMPRESSED or STREAM (the default).

FORMAT

MODE *mode*

PARAMETERS

mode

Specifies one of two values: COMPRESSED or STREAM (the default).

Restrictions

- The MODE command can only be used when connected to a remote host.
- Not all modes are supported by all remote hosts.

EXAMPLE

This example shows how to enable COMPRESSED mode.

```
FLOWERS.com>mode c
Type: Ascii (Non-Print), Structure: VMS, Mode: Compression
FLOWERS>COM>
```

MPUT

Copies multiple files from the local host to the remote host. MPUT is a synonym for MULTIPLE SEND. See MULTIPLE SEND for more information.

FORMAT

MPUT *files*

MULTIPLE DELETE

Deletes multiple files on the remote host. If you have turned on CONFIRM, (to confirm multiple transactions interactively), you are asked to confirm the deletion of each file. MULTIPLE DELETE is the same as MDELETE.

FORMAT

MULTIPLE DELETE *files*

PARAMETERS

files

Specifies which files to delete. Wildcard characters in files are expanded on the remote host.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the MULTIPLE DELETE command.

EXAMPLE

This example shows how to delete all files matching the remote wildcard * character.

```
FLOWERS.COM>multiple delete *.com;*
< List started
<Transfer completed.
<File deleted ok, file USERS:[FLOWERS.DOC.V32]LOGIN.COM;3.
<File deleted ok, file USERS:[FLOWERS.DOC.V32]LOGIN.COM;2.
<File deleted ok, file USERS:[FLOWERS.DOC.V32]LOGIN.COM;1.
```

MULTIPLE GET

Copies multiple files from the remote host to the local host. If you have turned on CONFIRM (to confirm multiple transactions interactively), you are asked to confirm the transfer of each file. MULTIPLE GET is the same as MGET and MULTIPLE RECEIVE.

FORMAT

MULTIPLE GET **files**

PARAMETERS

files

Specifies the names of the files to be copied. Wildcard characters are expanded on the remote host.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the MULTIPLE GET command.

EXAMPLE

This example shows how to transfer all files matching the * wildcard character.

```
FLOWERS.COM>multiple get *.com
<List started.
<Transfer completed.
<VMS retrieve of USERS:[HOLMES]COPY.COM;4 started.
<Transfer completed. 732 (8) bytes transferred.
<VMS retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
<Transfer completed. 2498 (8) bytes transferred.
```

MULTIPLE PUT

Copies multiple files from the local host to the remote host. MULTIPLE PUT is a synonym for MULTIPLE SEND. See MULTIPLE SEND for more information.

FORMAT

MULTIPLE PUT *files*

MULTIPLE RECEIVE

Copies multiple files from the remote host to the local host. **MULTIPLE RECEIVE** is a synonym for **MULTIPLE GET**. See **MULTIPLE GET** for more information.

FORMAT

MULTIPLE RECEIVE *files*

MULTIPLE SEND

Copies multiple files from the local host to the remote host. If you have turned on CONFIRM (to confirm multiple transactions interactively), you are asked to confirm the transfer of each file. MULTIPLE SEND is the same as MULTIPLE PUT and MPUT.

FORMAT

MULTIPLE SEND *files*

PARAMETERS

files

Specifies which files to copy. Wildcard characters in *files* are expanded on the local host.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the MULTIPLE SEND command.

EXAMPLE

This example shows how to transfer the files which match the "*.COM" wildcard.

```
YOYDYNE.COM>multiple send *.com
<VMS Store of ST_ROOT:[TMP]COPY.COM;4 started.
<Transfer completed. 732 (8) bytes transferred.
<VMS Store of ST_ROOT:[TMP]FIX.COM;3 started.
<Transfer completed. 496 (8) bytes transferred.
<VMS Store of ST_ROOT:[TMP]FOO.COM;11 started.
<Transfer completed. 436 (8) bytes transferred.
<VMS Store of ST_ROOT:[TMP]LOGIN.COM;4 started.
<Transfer completed. 2498 (8) bytes transferred.
YOYDYNE.COM>
```

OPEN

Establishes a connection to a host system. OPEN is a synonym for CONNECT. See CONNECT for more information.

FORMAT

OPEN *host*

PASSIVE

Enables or disables "passive" mode for file transfers with FTP servers on the opposite side of "firewall" gateways.

FORMAT

PASSIVE [*state*]

PARAMETERS

state

Specifies a value of ON, OFF, or TOGGLE.

DESCRIPTION

Typically, when an FTP client requests data from an FTP server, the server attempts to establish a connection with the client over which it transfers the data. If a "firewall" gateway separates the FTP client and server, the gateway may prohibit incoming connections. The solution is to enable "passive" mode transfers, in which the FTP server asks the FTP client to initiate the connection.

Note! Not all FTP servers support passive mode transfers.

The PASSIVE command lets you explicitly enable or disable passive mode. When you don't specify a state, the current state is toggled.

EXAMPLE

This example uses PASSIVE to allow the server to transfer a directory listing across a connection established by the FTP client rather than the server.

```
FTP>connect ftp.abc.com
Connection opened (Assuming 8-bit connections)
<HQ.ABC.COM MultiNet FTP Server Process 4.0(14) at Wed 8-Mar-00 10:57AM-
PST
HQ.ABC.COM>user anonymous
<anonymous user ok. Send real ident as password.
Password:*****
<Welcome to ABC's Anonymous FTP directory
<Guest User WHORFIN@FLOWERS.COM logged into USERS:[ANONYMOUS.ABC] at Wed
8-Mar-00 11:15AM-PST, job 208040a2.
<Directory and access restrictions apply
HQ.ABC.COM>passive on
[Passive mode is ON for transfers]
HQ.ABC.COM>dir
<List started.
FTP_ANON:[000000]
.INDEX;32      3      6-APR-2000 00:00 [WEBMASTER] R,RWED,RWED,R)
.WELCOME;4     2     16-MAR-2000 17:19 [WEBMASTER] R,RWED,RWED,R)
```

```

ABOUT.TXT;8          5   27-MAR-2000 14:54 [WEBMASTER] R,RWED,RWED,R)
COMPANY_INFORMATION.DIR;1|
                        1   3-JAN-2000 13:54 [WEBMASTER] (R,RWED,RWED,R)
CUSTOMER_SUPPORT.DIR;1
                        1   3-JAN-2000 13:55 [WEBMASTER] R,RWED,RWED,R)
                        544 27-MAR-2000 10:29 [WEBMASTER] R,RWED,RWED,R)
NFSACL.PS;1           72  27-MAR-2000 10:29 [WEBMASTER] R,RWED,RWED,R)
NFSACL.TXT;1          13  27-MAR-2000 10:29 [WEBMASTER] R,RWED,RWED,R)
PRODUCTS_AND_SERVICES.DIR;1
                        1   3-JAN-2000 13:58 [WEBMASTER] (R,RWED,RWED,R)
SERVER_MAP.TXT;54 60   6-APR-2000 00:04 [WEBMASTER] (RWED,RWED,RWED,R)
SET2048.MAR;2         5   27-MAR-2000 10:29 [WEBMASTER] (R,RWED,RWED,R)
THIRD_PARTY_TOOLS.DIR;1
                        1   3-JAN-2000 13:58 [WEBMASTER] (R,RWED,RWED,R)
UNZIP.EXE;3           155 27-MAR-2000 10:29 [WEBMASTER] R,RWED,RWED,R)
UNZIP_ALPHA.EXE;1 163 27-MAR-2000 10:29 [WEBMASTER] (R,RWED,RWED,R)
VMSIO.H;12            7   27-MAR-2000 10:29 [WEBMASTER] (R,RWED,RWED,R)
WHATS_NEW.TXT;1       1   5-MAR-2000 16:31 [WEBMASTER] (R,RWED,RWED,R)
Total of 1033 blocks in 15 files.
<Transfer completed.
HQ.ABC.COM>

```


PASSWORD

Sends a password to the remote FTP server explicitly, which normally happens automatically during login.

FORMAT

PASSWORD *password*

PARAMETERS

password

Specifies the password to send to the remote server. The password is not echoed when typed.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that the password be sent as part of the login procedure only.

EXAMPLE

This example shows how to send a password to the remote host.

```
FLOWERS.COM>pass airplane  
<Password accepted, thank you.  
FLOWERS.COM>
```

PORT

Specifies a TCP port number to use for the FTP control connection. Use this command only when connecting to an FTP server that provides a nonstandard FTP control connection port number.

FORMAT

PORT *port*

PARAMETERS

port

Specifies the port to use when establishing the FTP control connection with the remote server system.

EXAMPLE

This example shows how to explicitly specify a port number for the FTP control connection with the remote host.

```
FLOWERS.COM>port 1099  
FLOWERS.COM>
```

PROMPT-FOR-MISSING-ARGUMENTS

Turns on, off, or toggles (the default) whether or not FTP automatically prompts for missing command arguments.

FORMAT

PROMPT-FOR-MISSING-ARGUMENTS *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to use the PROMPT-FOR-MISSING-ARGUMENTS command.

```
FTP>prompt-for-missing-arguments
{Will NOT prompt for missing arguments};
SALES.FLOWERS.COM>get
?Missing remote filename
SALES.FLOWERS.COM>
```

PROMPT-ON-CONNECT

Turns on, off, or toggles (the default) whether or not FTP automatically prompts for a user name and password after making a connection.

FORMAT

PROMPT-ON-CONNECT *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to use **PROMPT-ON-CONNECT** to automatically prompt for a user name and password when a connection is made.

```
FTP>prompt-on-connect
[Will automatically prompt for username and password]
FTP>connect ftp.yod.com
Connection opened (Assuming 8-bit connections)
<FTP.YOD.COM MultiNet FTP Server Process 4.0(nn) at Fri 7-Apr-2000 7:42am
PST
Username: HOLMES
Password:
<User HOLMES logged into USERS:[HOLMES] at Fri 7-Apr-2000 14:42, job
2060011f.
FTP.YOD.COM>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command. If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

FORMAT

PUSH

PUT

Copies *local_file* on the local host to *remote_file* on the remote host. The current settings for type, mode, and structure are used during file transfers. PUT is the same as SEND.

FORMAT

PUT *local_file remote_file*

PARAMETERS

local-file

Specifies the name of the file on the local host.

remote-file

Specifies the name of the file on the remote host.

QUALIFIERS

/FDL

Puts a file in FDL format. When you create a file with the PUT /FDL qualifier, a file description language (FDL) file is created at the same time as the original file. The output file is converted to raw block format. When you retrieve a file with GET /FDL, the original format is restored using the attributes stored in the FDL file. If you do not use the /FDL qualifier with the GET command, the new raw block format is retained. In any case, the FDL file is retained and must be deleted independently. The /FDL qualifier provides compatibility with DEC TCP/IP Services for OpenVMS (formerly UCX). The FDL file has the same name except the string FDL is appended to the end of the file name.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the PUT command.

EXAMPLES

This example copies the file LOGIN.COM to the remote file FOO.COM.

```
FLOWERS.COM>put login.com foo.com
<VMS Store of ST_ROOT:[TMP]FOO.COM;12 started.
<Transfer completed. 2498 (8) bytes transferred.
FLOWERS.COM>
```

This example copies AFILE.TXT to BFILE.TXT and creates the additional BFILE.TXTFDL file. The BFILE.TXTFDL file is in ASCII format and is an appropriate FDL description of AFILE.TXT.

```
FLOWERS.COM>PUT /FDL AFILE.TXT BFILE.TXT
<ASCII Store of USERS:[HOLMES]BFILE.TXTFDL;1 started.
<Transfer completed. 888 (8) bytes transferred.
<IMAGE Store of USERS:[HOLMES]BFILE.TXT;1 started.
<Transfer completed. 6 (8) bytes transferred.
flowers.com
```

PWD

Displays the current working directory on the remote host. PWD is a synonym for SHOW-DIRECTORY. See SHOW-DIRECTORY for more information.

FORMAT

PWD

QUIT

Closes the current FTP connection and exits FTP. QUIT is a synonym for EXIT. See EXIT for more information.

FORMAT

QUIT

QUOTE

Sends a string to the FTP server verbatim. You can use QUOTE to access non-standard commands on the FTP server.

FORMAT

QUOTE *string*

PARAMETERS

string

Specifies a string to send to the server.

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

This example shows how to send a NOOP command to the remote host.

```
FLOWERS.COM>quote noop  
<NOOP command successful.  
FLOWERS.COM>
```

RECEIVE

Copies *remote-file* from the remote host to *local-file* on the local host. The current settings for type, mode, and structure are used during file transfers. **RECEIVE** is a synonym for **GET**.

FORMAT

RECEIVE *remote-file* [*local-file*]

PARAMETERS

remote-file

Specifies the name of the file on the remote host.

local-file

Specifies the name of the file on the local host.

QUALIFIERS

/FDL

Gets a file previously saved with the **PUT /FDL** command. When you create a file with the **PUT /FDL** qualifier, a file description language (FDL) file is created at the same time as the original file. The output file is converted to raw block format. When you retrieve a file with **RECEIVE /FDL**, the original format is restored using the attributes stored in the FDL file. If you do not use the **/FDL** qualifier with the **RECEIVE** command, the new raw block format is retained. In any case, the FDL file is retained and must be deleted independently. The **/FDL** qualifier provides compatibility with DEC TCP/IP Services for OpenVMS (formerly UCX). The FDL file has the same name except the string **FDL** is appended to the end of the file name.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the **GET** command.

EXAMPLE

This example shows how to transfer a file to the local host.

```
FLOWERS.COM>receive login.com
To local file: RETURN
<VMS retrieve of USERS:[HOLMES]LOGIN.COM;1 started.
<Transfer completed. 2498 (8) bytes transferred.
FLOWERS.COM>
```

RECORD-SIZE

Sets or displays the record size for IMAGE mode transfers.

FORMAT

RECORD-SIZE [*size*]

PARAMETERS

size

Specifies the record size for IMAGE mode transfers. Values range from 1 to 32767. When omitted, the current setting is displayed. The default record size is 512 bytes.

EXAMPLE

```
$ ftp ftp.yod.com
FTP.YOD.COM MultiNet FTP user process 4.3(nnn)
Connection opened (Assuming 8-bit connections)
<FTP.YOD.COM MultiNet FTP Server Process 4.0(nnn) at Fri 7-Apr-2000
7:42am-PST
FTP>record 1024
FTP>record
Record size for IMAGE files: 1024
FTP>
```

REMOTE-HELP

Displays information about commands available on the FTP server.

FORMAT

REMOTE-HELP

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

This example shows how to retrieve help from a remote host.

```
UNIX.FLOWERS.COM>remote-help
<The following commands are recognized (* =>'s unimplemented).
< USER      PORT      STOR      MSAM*     RNT0      NLST      MKD       CDUP
< PASS       PASV      APPE      MRSQ*     ABOR      SITE      XMKD      XCUP
< ACCT*      TYPE      MLFL*     MRCP*     DELE      SYST      RMD       STOU
< SMNT*      STRU      MAIL*     ALLO      CWD       STAT      XRMD      SIZE
< REIN*      MODE      MSND*     REST      XCWD      HELP      PWD       MDTM
< QUIT       RETR      MSOM*     RNFR      LIST      NOOP      XPWD
<Direct comments to ftp-bugs@ucbarpa.Berkeley.EDU.
UNIX.FLOWERS.COM>
```

REMOVE-DIRECTORY

Deletes a directory on the remote host. REMOVE-DIRECTORY is the same as RMDIR.

FORMAT

REMOVE-DIRECTORY *dir*

PARAMETERS

dir

Specifies the name of the directory to be removed.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you be logged in before using the REMOVE-DIRECTORY command.

EXAMPLE

This example shows how to delete the "test" subdirectory from the remote host.

```
FLOWERS.COM>remove-directory test  
<"USERS:[HOLMES.TEST]" Directory deleted  
FLOWERS.COM>
```

RENAME

Renames files on the remote host.

FORMAT

RENAME *file1 file2*

PARAMETERS

file1

Specifies the name of the file to be renamed.

file2

Specifies the new name of *file1*.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the RENAME command.

EXAMPLE

This example shows how to rename COPY.COM to NEWCOPY.COM on the remote host.

```
FLOWERS.COM>rename Copy.com newcopy.com  
<Old FILE renamed to USERS:[HOLMES]NEWCOPY.COM;1.  
FLOWERS.COM>
```

RETAIN

Turns on, off, or toggles (the default) the retention of OpenVMS version numbers in file transfers. By default, version numbers are stripped from OpenVMS file names before they are sent over the network.

FORMAT

RETAIN mode

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to enable retention of OpenVMS version numbers.

```
FTP>retain  
[Transferred files will retain their version numbers]  
FTP>
```


RM

Deletes a file on the remote host. RM is a synonym for DELETE. See DELETE for more information.

FORMAT

RM *file*

RMDIR

Deletes a directory on the remote host. RMDIR is a synonym for REMOVE-DIRECTORY. See REMOVE-DIRECTORY for more information.

FORMAT

RMDIR *dir*

SEND

Copies *local_file* on the local host to *remote_file* on the remote host. The current settings for type, mode, and structure are used during file transfers. SEND is the same as PUT.

FORMAT

SEND *local_file remote_file*

PARAMETERS

local_file

Specifies the name of the file on the local host to be copied.

remote_file

Specifies the destination file name on the remote host.

QUALIFIERS

/FDL

Sends a file in FDL format. When you create a file with the SEND /FDL qualifier, a file description language (FDL) file is created at the same time as the original file. The output file is converted to raw block format. When you retrieve a file with GET /FDL, the original format is restored using the attributes stored in the FDL file. If you do not use the /FDL qualifier with the GET command, the new raw block format is retained. In any case, the FDL file is retained and must be deleted independently. The /FDL qualifier provides compatibility with DEC TCP/IP Services for OpenVMS (formerly UCX). The FDL file has the same name except the string FDL is appended to the end of the file name.

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the SEND command.

EXAMPLE

This example shows how to transfer the file LOGIN.COM to the remote file FOO.COM.

```
FLOWERS.COM>send login.com foo.com  
<VMS Store of ST_ROOT:[TMP]FOO.COM;12 started.  
<Transfer completed. 2498 (8) bytes transferred.
```

SET

Sets automatic login information for host.

FORMAT

SET *host*

PARAMETERS

host

Specifies the host for which you want to set automatic login information.

QUALIFIERS

/USER:username

Specifies the user name sent when a connection is made to *host*.

/PASSWORD:password

Specifies the password sent when a connection is made to *host*.

/ACCOUNT:account

Specifies the account is sent when a connection is made to *host*.

DESCRIPTION

When a connection to *host* is made, FTP uses the information set to automatically log in. This command is usually used in the FTP.INIT file to specify a list of hosts and their login information. If FTP.INIT contains passwords in clear text, it is imperative that you protect the file from access by other users. If you specify /USER but not /PASSWORD, an automatic login is attempted and, if necessary, a password prompt displayed.

Restrictions

Do not use this command when connected to a remote host.

USAGE NOTE

If you do not specify any qualifiers, any automatic login information is cleared.

EXAMPLE

This example sets the user name and password for the host DS.INTERNIC.NET.

```
FTP>ds.internic.net /user:anonymous /pass:guest
```

SHOW-DIRECTORY

Displays the current working directory on the remote host. SHOW DIRECTORY is the same as PWD.

FORMAT

SHOW-DIRECTORY

Restrictions

- Use this command only when connected to a remote host.
- Most remote hosts require that you log in before using the SHOW-DIRECTORY command.

EXAMPLE

This example shows how to retrieve the remote default directory.

```
FLOWERS.COM>show  
<"ST_ROOT: [TMP]" is current directory.  
FLOWERS.COM>
```

SITE

Specifies commands that are interpreted by the MultiNet FTP server for use on the server host.

FORMAT

SITE *command*

PARAMETERS

command

Selects a command from the following:

RMS RECSIZE <i>n</i>	Indicates a non-default record size for files transferred in IMAGE mode to the FTP server. Record size values can range from 1 to 32767; the default is 512 bytes.
SPAWN	Allows users to execute commands on the server host. The command must not require a terminal device, and must exit on completion. You cannot use this command during an anonymous FTP session.

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

FORMAT

SPAWN [*command*]

PARAMETERS

command

Specifies a command to execute. If you omit *command*, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=file-spec

Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES

/NOLOGICAL_NAMES

Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS

/NOSYMBOLS

Specifies that global and local names are not passed to the subprocess.

/WAIT

/NOWAIT

Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=file-spec

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

STATISTICS

Turns on, off, or toggles (the default) STATISTICS mode. In STATISTICS mode, FTP displays, upon completion of file transfers, timing statistics about the transfer.

If the logical MULTINET_FTP_STATISTICS_IN_HHMMSS is defined with either 1, T, or Y, then the elapsed time displays in HH:MM:SS format if statistics are requested using the STATISTICS mode.

FORMAT

STATISTICS *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to enable STATISTICS mode.

```
FTP>statistics  
[Transfer statistics printing is ON]  
FTP>
```


STATUS

Displays the status of the FTP server.

FORMAT

STATUS [*data*]

PARAMETERS

data

Sends this command data to the FTP server; data depends on the implementation of the FTP server. This parameter is optional.

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

```
FLOWERS.COM>status
<FLOWERS.COM MultiNet FTP Server Process 4.3(nnn)
<User HOLMES is logged into directory ST_ROOT:[TMP]
<The current transfer parameters are:
<   MODE S
<   Stru O VMS
<   TYPE A N
<A connection is open to host FLOWERS.COM
<The data connection is CLOSED.
FLOWERS.COM
```

STREAM

Turns on, off, or toggles (the default) the creation of binary output files as Stream_LF files.

FORMAT

STREAM *mode*

PARAMETERS

mode

Specifies ON, OFF, or TOGGLE.

EXAMPLE

```
FLOWERS.COM>stream  
[ IMAGE files will be written as Stream_LF format]  
FLOWERS.COM>
```

STRUCTURE

Sets the transfer structure to *structure*.

FORMAT

STRUCTURE *structure*

PARAMETERS

structure

Specifies a value of FILE, RECORD, or VMS.

- Use FILE (the default) when connecting to systems that do not support VMS structure negotiation.
- Use RECORD to transfer files when you want to preserve the record boundaries.
- Use VMS to transfer files with arbitrary RMS attributes transparently. Transparent transfer is negotiated automatically between systems that support it. RMS semantics are passed along with the data.

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

```
FLOWERS.COM>stru r  
Type:Ascii (Non-Print), Structure: Record, Mode: Stream  
FLOWERS.COM>
```

TAKE

Interprets FTP commands in a file. When the end of the file is encountered, the FTP command interpreter returns to its previous input source. You can nest TAKE commands up to ten levels deep.

FORMAT

TAKE *file*

PARAMETERS

file

Specifies the name of the file that contains commands to be interpreted.

EXAMPLE

This example shows how to take commands from the file FTP.COMMANDS.

```
FTP>take ftp.commands
```

TENEX

Changes the byte size for transferring binary files to or from a TOPS-20 system.

FORMAT

TENEX

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

This example shows how to set the transfer type to TENEX.

```
FLOWERS.COM>tenex
```

```
Type: Logical-Byte (Byte Size 8), Structure: File, Mode: Stream
```

```
FLOWERS.COM>
```

TYPE

Sets the transfer type to *type*.

FORMAT

TYPE *type*

PARAMETERS

type

Specifies a value of ASCII, BACKUP, BINARY, IMAGE, or LOGICAL-BYTE.

- Use TYPE ASCII (the default) for transferring text files.
- Use TYPE BACKUP to set the transfer type to IMAGE and write the local file with 2048-byte fixed length records. Use this command to transfer VAX/VMS BACKUP save sets.
- Use TYPE BINARY to transfer binary files (same as TYPE IMAGE).
- Use TYPE IMAGE to transfer binary files (for example, .EXE).
- Use TYPE LOGICAL-BYTE to transfer binary files to or from a TOPS-20 machine.

Restrictions

Use this command only when connected to a remote host.

EXAMPLE

This example shows how to set the type to transfer an image file.

```
FLOWERS.COM>type i  
Type: Image, Structure: File, Mode: Stream  
FLOWERS.COM>
```

USER

Identifies you to the remote FTP server. **USER** is a synonym for **LOGIN**. See **LOGIN** for more information.

FORMAT

USER *user* [*password*]

VERBOSE

Turns on, off, or toggles (the default) VERBOSE mode. VERBOSE mode causes FTP to display all responses from the remote FTP server as they are received.

FORMAT

VERBOSE *mode*

PARAMETERS

mode

Specifies a value of ON, OFF, or TOGGLE.

EXAMPLE

This example shows how to enable VERBOSE mode.

```
FTP>verbose  
[Verbose reply printing is ON]  
FTP>
```


VERSION

Prints information about the FTP program version.

FORMAT

VERSION

EXAMPLE

This example shows how to print the FTP program version number.

```
FLOWERS.COM>version  
FLOWERS.COM MultiNet FTP user process 4.3(nnn)  
FLOWERS.COM>
```


Appendix C

TELNET Command Reference

The MultiNet TELNET utility uses the Internet-standard TELNET protocol to establish a virtual terminal connection between your terminal and a remote host. This appendix lists the commands you can use during a TELNET session.

Command Summary

The following table lists the TELNET commands:

Table C-1 TELNET Command Summary

Command:	Description:
ABORT	Sends an ABORT OUTPUT sequence to the remote host.
ATTACH	Detaches the terminal from the calling process and reattaches it to another process.
ATTN	Sends an INTERRUPT PROCESS sequence to the remote host.
AYT	Sends an ARE YOU THERE sequence to the remote host.
BINARY	Attempts to negotiate binary (8-bit) mode with the remote system.
BREAK	Sends a BREAK sequence to the remote host.
BYE	Closes any open TELNET connection and exits to DCL.
CLOSE	Closes the TELNET connection.

Table C-1 TELNET Command Summary (Continued)

Command:	Description:
CONNECT	Establishes a TELNET connection to a host.
CREATE-NTY	Connects the local end of a TELNET connection to an NTY pseudo-terminal device.
DEBUG	Displays TELNET option negotiations.
ECHO	Turns on or off remote host character echoing.
EXIT	Closes any open TELNET connection and exits to DCL. EXIT is the same as BYE and QUIT.
HELP	Displays help information for the specified TELNET command.
LOG-FILE	Enables or disables logging of the TELNET session.
PUSH	Starts and attaches a DCL subprocess.
QUIT	Closes any open TELNET connection and exits to DCL. QUIT is the same as EXIT.
SET ABORT-OUTPUT-CHARACTER	Sets the character that TELNET maps to the ABORT OUTPUT sequence.
SET ARE-YOU-THERE-CHARACTER	Sets the character that TELNET maps to the ARE YOU THERE sequence.
SET AUTO-FLUSH	Turns auto-flushing on or off.
SET BREAK-CHARACTER	Sets the character that TELNET maps to the BREAK sequence.
SET DEBUG	Enables or disables the display of TELNET option negotiations.
SET ERASE-CHARACTER-CHARACTER	Sets the character that TELNET maps to the ERASE CHARACTER sequence.
SET ERASE-LINE-CHARACTER	Sets the character that TELNET maps to the ERASE LINE sequence.
SET-ESCAPE-CHARACTER	Sets the character that switches TELNET to command mode.
SET EXTENDED	Causes TELNET to go into extended command mode automatically whenever you type the TELNET ESCAPE character, Ctrl / ^ by default.

Table C-1 TELNET Command Summary (Continued)

Command:	Description:
SET INTERRUPT-PROCESS-CHARACTER	Sets the character that TELNET maps to the INTERRUPT PROCESS sequence.
SET LOCAL-FLOW-CONTROL	Specifies whether or not Ctrl/S and Ctrl/Q should be treated by the local terminal driver as XON and XOFF.
SET LOG-FILE	Enables or disables logging of the TELNET session.
SET REMOTE-USERNAME	Specifies the user name to which you wish to log in using Kerberos.
SET UNIX-LINE-TERMINATOR	Causes TELNET to use the 4.3BSD UNIX end-of-line specification, Ctrl/NULL .
SPAWN	Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH.
STATUS	Displays the status of the current TELNET connection and parameters.
TEMRINAL-TYPE	Specifies a terminal type for the TELNET session.
VERSION	Displays the TELNET version number.

ABORT

Sends an ABORT OUTPUT sequence to the remote host. If the remote host is running MultiNet, the TELNET ABORT OUTPUT sequence is treated as a **Ctrl/O**.

FORMAT

ABORT

Restrictions

Use this command only in extended mode.

EXAMPLE

This example sends the ABORT OUTPUT sequence to the remote system.

```
TELNET>abort
```

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the SPAWN SHOW PROCESS /SUBPROCESSES command to list the names of subprocesses. Use the DCL LOGOUT command to return to the original process. If the MULTINET_DISABLE_SPAWN logical is enabled, ATTACH does not work.

FORMAT

ATTACH *process-name*

PARAMETERS

process_name

Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

ATTN

Sends an INTERRUPT PROCESS sequence to the remote host. If the remote host is also running MultiNet, the TELNET INTERRUPT PROCESS sequence is treated as a **Ctrl/C**.

FORMAT

ATTN

Restrictions

Use this command only in extended mode.

EXAMPLE

This example sends the INTERRUPT PROCESS sequence to the remote system.

```
TELNET>attn
```


AYT

Sends an ARE YOU THERE sequence to the remote host. If the remote host is also running MultiNet, the ARE YOU THERE sequence is treated as a **Ctrl/T**.

Note! AYT does not work if the terminal is not enabled for broadcasts. Invoke the DCL command SET TERMINAL /BROADCAST before using AYT if broadcasts have been disabled.

FORMAT

AYT

EXAMPLE

This example shows how to ensure the host is still active.

```
TELNET>ayt
```

```
FNORD: :_VTA81: 01:37:57 (DCL) CPU=00:00:01.83 PF=2298 IO=530 MEM=345
```

BINARY

Attempts to negotiate binary (8-bit) mode with the remote system.

FORMAT

BINARY

Restrictions

Use this command only in extended mode.

EXAMPLE

```
TELNET>binary
```

BREAK

Sends a BREAK sequence to the remote host. If the remote host is running MultiNet, the BREAK sequence is treated as a **Ctrl/C**.

FORMAT

BREAK

Restrictions

Use this command only in extended mode.

EXAMPLE

```
TELNET>break
```

BYE

Closes any open TELNET connection and exits to DCL. BYE is the same as EXIT.

FORMAT

BYE

EXAMPLE

```
TELNET>bye  
$
```

CLOSE

Closes the TELNET connection.

FORMAT

CLOSE

USAGE NOTES

If you specified the remote host in the DCL TELNET command, exit to DCL. If you connected to the remote host in TELNET command mode, return to general command mode.

On most remote hosts, closing the connection is seen as a modem-style terminal hangup. If the remote host is also running MultiNet and OpenVMS virtual terminals are enabled, the remote login session becomes detached.

Restrictions

Use this command only in extended mode.

EXAMPLE

```
TELNET>close
```

CONNECT

Establishes a TELNET connection to a host. TELNET connections may be established using NETWARE or INTERNET protocols; the default is INTERNET.

FORMAT

CONNECT [*protocol*] *host* [*port*]

PARAMETERS

protocol

Specifies the protocol to use to establish the connection. The protocol can be NETWARE or INTERNET (the default).

host

Specifies the host to which to establish the connection. With the INTERNET protocol, the host can be a name or a numeric IP address. With the NETWARE protocol, you must specify a name.

port

Specifies the remote port number or name to use for the connection. With the INTERNET protocol, the default is the TELNET port. With the NETWARE protocol, the port specification is not an option.

Restrictions

Do not use this command in extended mode.

EXAMPLE

This example shows how to connect to a remote system.

```
TELNET>connect internet unix  
Trying... Connected to UNIX.FLOWERS.COM, a VAXSTATION-II running UNIX.4.3  
BSD UNIX (unix.FLOWERS.com)  
login:
```

CREATE-NTY

Connects the local end of a TELNET connection to an NTY pseudo-terminal device. This device can be used by other applications such as KERMIT. This command includes the remote host and port number in the SHOW TERMINAL “remote port information” field.

FORMAT

CREATE-NTY

EXAMPLE

```
TELNET>create-nty  
TELNET session now connected to _NTY3:  
%DCL-I-ALLOC, _NTY3: allocated  
$
```

DEBUG

Displays TELNET option negotiations.

FORMAT

DEBUG *[mode]*

PARAMETERS

mode

Specifies whether debugging is enabled (default) or disabled (OFF). Debug mode causes TELNET to display option negotiations between the local host and the foreign host.

EXAMPLE

This example shows how to enable DEBUG mode.

```
TELNET>debug on
```


ECHO

Turns on or off remote host character echoing.

FORMAT

ECHO *mode*

PARAMETERS

mode

Specifies whether the server handles character echoing. If you specify OFF, TELNET performs local character echoing. If you specify ON, the remote system performs the echoing.

Restrictions

Use this command only in extended mode.

EXAMPLE

```
TELNET>echo off
```

EXIT

Closes any open TELNET connection and exits to DCL. EXIT is the same as BYE and QUIT.

FORMAT

EXIT

EXAMPLE

This example shows how to exit TELNET.

```
TELNET>exit  
$
```

HELP

Displays help information for the specified TELNET command. Type **HELP ?** to see a list of **HELP** topics, or type **HELP** with no argument to see general information regarding TELNET.

FORMAT

HELP *[command]*

PARAMETERS

command

Specifies information about this command.

LOG-FILE

Enables or disables logging of the TELNET session. If you specify *log_file*, everything received by the local system from the remote system is copied into this file.

FORMAT

LOG-FILE *log_file*

PARAMETERS

log_file

Specifies a file to which to write a log of the TELNET session. If you don't specify a file, logging is enabled to the file TELNET.LOG. If you specify the file name NONE, logging is disabled.

Restrictions

LOG-FILE is not supported in 3270 or 5250 modes.

EXAMPLE

This example shows how to enable TELNET output to be logged to the file ST_TMP:FNORD.LOG.

```
TELNET>log-file st_tmp:fnord.log
[Log file open (ST_TMP:<TMP>FNORD.LOG.1)]
TELNET>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command. If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

FORMAT

PUSH

QUIT

Closes any open TELNET connection and exits to DCL. QUIT is the same as EXIT.

FORMAT

QUIT

EXAMPLE

This example shows how to exit TELNET.

```
TELNET>quit  
$
```

SET ABORT-OUTPUT-CHARACTER

Sets the character that TELNET maps to the ABORT OUTPUT sequence. The value set by this command is not the character passed to the remote host. The remote host receives an ABORT OUTPUT sequence; SET ABORT-OUTPUT-CHARACTER defines the key you press to tell TELNET to send an ABORT OUTPUT sequence. This character can also be set by invoking TELNET with the /ABORT_OUTPUT_CHARACTER qualifier.

FORMAT

SET ABORT-OUTPUT-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the ABORT OUTPUT sequence to the TELNET server.

If you type the command without specifying character, it defaults to **Ctrl/O**.

EXAMPLE

This example sets the ABORT OUTPUT character to **Ctrl/A**.

```
TELNET>set abort "^A"  
[Abort Output character set to ^A]  
TELNET>
```

SET ARE-YOU-THERE-CHARACTER

Sets the character that TELNET maps to the ARE YOU THERE sequence. The value set by this command is not the character passed to the remote host. The remote host receives an ARE YOU THERE sequence; SET ARE-YOU-THERE-CHARACTER defines the key you press to tell TELNET to send an ARE YOU THERE sequence. This character can also be set by invoking TELNET with the /ARE_YOU_THERE_CHARACTER qualifier. The ARE YOU THERE sequence can be sent by pressing the ARE YOU THERE character or by issuing the TELNET AYT command.

Note! The ARE YOU THERE sequence only displays an information line from the host if broadcasts are enabled for the terminal.

FORMAT

SET ARE-YOU-THERE-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the ARE YOU THERE sequence to the TELNET server.

If you type the command without specifying character, it defaults to **Ctrl/T**.

EXAMPLE

This example sets the ARE YOU THERE character to **Ctrl/T**.

```
TELNET>set are-you-there "^T"  
[Are-You-There character set to ^T]  
TELNET>
```


SET AUTO-FLUSH

Turns auto-flushing on or off. You can also set this mode by invoking TELNET with the /AUTOFLUSH qualifier.

When you define an ABORT-OUTPUT character, enabling AUTO-FLUSH (SET AUTO-FLUSH ON) causes TELNET to flush any data which may be in the network buffers when the ABORT-OUTPUT character is typed. The TELNET client does this by sending a TIMING-MARK command to the TELNET server and discarding all data until one is received in response.

FORMAT

SET AUTO-FLUSH *mode*

PARAMETERS

mode

Turns auto-flush ON or OFF. If you do not specify *mode*, it defaults to ON.

EXAMPLE

This example sets the Auto Flush option on.

```
TELNET>set auto-flush on
TELNET>
```

SET BREAK-CHARACTER

Sets the character that TELNET maps to the BREAK sequence. The value set by this command is not the character passed to the remote host. The remote host receives a BREAK sequence; SET BREAK-CHARACTER defines the key you press to tell TELNET to send a BREAK sequence. You can also set this character by invoking TELNET with the /BREAK_CHARACTER qualifier.

FORMAT

SET BREAK-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the BREAK sequence to the TELNET server.

If you type the command without specifying *character*, it defaults to **Ctrl/A**.

EXAMPLE

This example sets the BREAK character to **Ctrl/A**.

```
TELNET>set break "^A"  
[Break character set to ^A]  
TELNET>
```

SET DEBUG

Enables or disables the display of TELNET option negotiations. You can also set this mode by invoking TELNET with the /DEBUG qualifier.

FORMAT

SET DEBUG [*mode*]

PARAMETERS

mode

Turns debugging ON or OFF. If *mode* is not specified, the default is ON.

EXAMPLE

This example enables DEBUG mode.

```
TELNET>set debug on
```

SET ERASE-CHARACTER-CHARACTER

Sets the character that TELNET maps to the ERASE CHARACTER sequence. The value set by this command is not the character passed to the remote host. SET ERASE-CHARACTER-CHARACTER defines the key you press to tell TELNET to send an ERASE CHARACTER sequence. This character can also be set by invoking TELNET with the /ERASE_CHARACTER_CHARACTER qualifier.

FORMAT

SET ERASE-CHARACTER-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the ERASE CHARACTER sequence to the TELNET server.

If you type this command without specifying character, it defaults to **DEL**.

EXAMPLE

This example sets the ERASE CHARACTER to **Ctrl/A**.

```
TELNET>set erase "^A"  
[Erase character set to "^A"]  
TELNET>
```

SET ERASE-LINE-CHARACTER

Sets the character that TELNET maps to the ERASE LINE sequence. The value set by this command is not the character passed to the remote host; SET ERASE-LINE-CHARACTER defines the key you press to tell TELNET to send an ERASE LINE sequence. This character can also be set by invoking TELNET with the /ERASE_LINE_CHARACTER qualifier.

FORMAT

SET ERASE-LINE-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the ERASE LINE sequence to the TELNET server.

If you type the command without specifying character, it defaults to **Ctrl/U**.

EXAMPLE

This example sets the ERASE LINE character to **Ctrl/U**.

```
TELNET>set erase-line "^U"  
[Escape Line character set to ^U  
TELNET>
```

SET ESCAPE-CHARACTER

Sets the character that switches TELNET to command mode. This character can also be set by invoking TELNET with the /ESCAPE_CHARACTER qualifier.

FORMAT

SET ESCAPE-CHARACTER *character*

PARAMETERS

character

Specifies which character is used as the TELNET ESCAPE character.

If you type the command without specifying character, it defaults to **Ctrl/^**.

EXAMPLE

This example sets the ESCAPE character to **Ctrl/^**.

```
TELNET>set escape "^\  
[Escape character set to ^\  
TELNET>
```

SET EXTENDED

Causes TELNET to go into extended command mode automatically whenever you type the TELNET ESCAPE character, **Ctrl**/**^** by default.

FORMAT

SET EXTENDED *mode*

PARAMETERS

mode

Turns extended mode ON or OFF. If you do not specify *mode*, it defaults to ON.

EXAMPLE

This example enables the extended option.

```
TELNET>set extended on  
TELNET>
```

SET INTERRUPT-PROCESS-CHARACTER

Sets the character that TELNET maps to the INTERRUPT PROCESS sequence. The value set by this command is not the character passed to the remote host. The remote host receives an INTERRUPT PROCESS sequence; SET INTERRUPT-PROCESS-CHARACTER defines the key you press to tell TELNET to send an INTERRUPT PROCESS sequence. You can also set this character by invoking TELNET with the /INTERRUPT_PROCESS_CHARACTER qualifier.

FORMAT

SET INTERRUPT-PROCESS-CHARACTER *character*

PARAMETERS

character

Specifies which character sends the INTERRUPT PROCESS sequence to the TELNET server.

If you type the command without specifying character, it defaults to **Ctr1/C**.

EXAMPLE

This example sets the INTERRUPT PROCESS character to **Ctr1/C**.

```
TELNET>set interrupt-process "^C"  
[Interrupt Process character set to ^C]  
TELNET>
```


SET LOCAL-FLOW-CONTROL

Specifies whether or not **Ctr1/s** and **Ctr1/Q** should be treated by the local terminal driver as XON and XOFF. You can also set this mode by invoking TELNET with the **/LOCAL_FLOW_CONTROL** qualifier.

Use of this qualifier causes a more responsive XOFF, which helps prevent data loss, but the remote system is unable to see any **Ctr1/s** characters.

The default under the MultiNet TELNET utility is to use the current setting of the VMS terminal characteristic **TT\$_TTSYNC** (set by the DCL command **SET TERMINAL/TTSYNC**), unless the remote host supports the **TOGGLE-FLOW-CONTROL** TELNET option. In that case, the **LOCAL-FLOW-CONTROL** option is set automatically by the TELNET server.

FORMAT

SET LOCAL-FLOW-CONTROL *mode*

PARAMETERS

mode

Turns local flow control ON or OFF. If *mode* is not specified, it defaults to ON.

EXAMPLE

This example enables local processing of **Ctr1/s** and **Ctr1Q**.

```
TELNET>set local-flow on
TELNET>
```

SET LOG-FILE

Enables or disables logging of the TELNET session. You can also set a log file by invoking TELNET with the /LOG_FILE qualifier.

FORMAT

SET LOG-FILE *log_file*

PARAMETERS

log_file

Specifies a file to which to write the log of the TELNET session. If you specify *log_file*, everything received by the local system from the remote system is copied into this file. If you don't specify a file, logging is enabled to the file TELNET.LOG. If you specify the file name NONE, logging is disabled.

Restrictions

log_file is not supported in 3270 and 5250 modes.

SET REMOTE-USERNAME

Specifies the user name to which you wish to log in using Kerberos. If you are not logging in with the /AUTH qualifier, TELNET prompts you to supply a user name.

FORMAT

SET REMOTE-USERNAME *username*

PARAMETERS

username

Specifies the user name to which you wish to log in using Kerberos.

SET UNIX-LINE-TERMINATOR

Causes TELNET to use the 4.3BSD UNIX end-of-line specification, **Ctrl/NULL**. You can also set this mode by invoking TELNET with the **/UNIX** qualifier. This command is useful when using TELNET to connect to 4.3BSD UNIX systems whose TELNET server does not conform to the TELNET specification.

FORMAT

SET UNIX-LINE-TERMINATOR *mode*

PARAMETERS

mode

If *mode* is ON, TELNET uses the 4.3BSD UNIX end-of-line specification, **Ctrl/NULL**.

If *mode* is OFF (the default), TELNET uses the standard end-of-line specification, **Ctrl/LF**.

EXAMPLE

This example enables use of a 4.3BSD UNIX-style line terminator.

```
TELNET>set unix-line-terminator on
TELNET>
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

FORMAT

SPAWN [*command*]

PARAMETERS

command

Specifies a command to execute. If you omit command, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=file-spec

Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES

/NOLOGICAL_NAMES

Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS

/NOSYMBOLS

Specifies that global and local names are not passed to the subprocess.

/WAIT

/NOWAIT

Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=file-spec

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

STATUS

Displays the status of the current TELNET connection and parameters.

FORMAT

STATUS

EXAMPLE

```
TELNET>status
This is FNORD.FOO.COM, VMS Version V6.0
Connected to host CONE.FOO.COM, a VAXSTATION-4000-90 running VMS via TCP.
Remote host is echoing
Host is not sending binary
Client is not sending binary
NO Abort Output character
NO Interrupt Process character
NO Are-You-There character
NO Break Character character
NO Erase Character character
NO Erase Line character
Escape Character character is '^'
Normal End Of Line mapping
Local Flow control
No log file
Terminal type is vt100
Remote host status reply:
FNORD::_VTA12: 16:40:02 (DCL) CPU=00:00:03.21 PF=686 IO=196 MEM=514
```

TERMINAL-TYPE

Specifies a terminal type for the TELNET session.

FORMAT

TERMINAL-TYPE *type*

PARAMETERS

type

Refer to RFC-1340 for a list of possible terminal types. RFCs are provided on the MultiNet CD-ROM. MultiNet TELNET permits you to specify any terminal type, even if the terminal type is not listed in the RFC. The TERMINAL-TYPE command has the same effect as invoking TELNET with the /TERMINAL_TYPE qualifier.

EXAMPLE

```
TELNET>terminal-type dec-vt220
```

VERSION

Displays the TELNET version number.

FORMAT

VERSION

EXAMPLE

```
TELNET>version  
This is MultiNet TELNET-32 Version 4.3(nnn)  
TELNET>
```


Appendix D

TFTP Command Reference

The MultiNet TFTP utility uses the Internet-standard Trivial File Transfer Protocol (TFTP) to transfer files between the local host and a remote host. This appendix describes the commands you can use during a TFTP session.

Command Summary

The following table lists the TFTP commands:

Table D-1 TFTP Command Summary

Command	Description
CONNECT	Specifies the name or address of the TFTP server.
GET	Transfers remote_file on the remote host to local_file on the local host.
PUT	Copies local_file on the local host to remote_file on the remote host.
QUIT	Terminates TFTP and returns to DCL.
REXMIT	Specifies the amount of time TFTP waits for a response to arrive before retransmitting a request. The default value for the retransmission timer is five seconds.
STATUS	Displays the current TFTP status.
TIMEOUT	Sets the amount of time TFTP waits for a response from the server before aborting a transfer.
TRACE	Toggles TFTP packet tracing.

CONNECT

Specifies the name or address of the TFTP server. This value overrides the command line host specification. You may use either a symbolic host name or an Internet address.

This command does not cause any network action, but sets the destination address for the TFTP UDP packets. If the host cannot be reached, an error is not displayed until a GET or PUT command is attempted.

FORMAT

connect *host*

PARAMETERS

host

Specifies a remote host.

EXAMPLE

This example connects to the host FLOWERS.COM.

```
tftp>connect flowers.com
```

GET

Transfers *remote_file* on the remote host to *local_file* on the local host.

You must specify an absolute path name (device, directory, and file name) for *remote_file*, and typically the server requires the file to be world-readable. If you do not specify *local_file*, the default is the same name and directory as *remote_file*.

FORMAT

```
get remote_file [local_file]
```

PARAMETERS

remote_file

Specifies the name of the input file on the remote host.

local_file

Specifies the name of the output file on the local host.

EXAMPLE

This example retrieves the file `USERS:[SMITH]LOGIN.COM` and stores it in the file `LOGIN.COM`.

```
tftp>get users:[smith]login.com login.com  
Received 2361 bytes in 1 seconds.  
tftp>
```

PUT

Copies *local_file* on the local host to *remote_file* on the remote host.

You must use absolute pathnames on *remote_file*, and typically the server requires the file to already exist and be world-writable (W:W). If you do not specify *remote_file*, it defaults to the same name and directory as *local_file*.

FORMAT

put *local_file* [*remote_file*]

PARAMETERS

local_file

Specifies the name of the input file on the local host.

remote_file

Specifies the name of the output file on the remote host.

EXAMPLE

This example transfers SYS\$LOGIN:LOGIN.COM to the remote file specification "/tmp/foo".

```
tftp>put sys$login:login.com /tmp/foo
Sent 2361 bytes in 1 second.
tftp>
```

QUIT

Terminates TFTP and returns to DCL.

FORMAT

quit

EXAMPLE

```
tftp>quit  
$
```

REXMT

Specifies the amount of time TFTP waits for a response to arrive before retransmitting a request. The default value for the retransmission timer is five seconds.

FORMAT

rexmt *seconds*

PARAMETERS

seconds

Sets the TFTP retransmission timer to the specified number of seconds.

EXAMPLE

This example sets the TFTP retransmission timer to 10 seconds.

```
tftp>rexmt 10
```

STATUS

Displays the current TFTP status.

FORMAT

STATUS

EXAMPLE

This example shows how to display TFTP status after a connection has been made to FLOWERS.COM. All values shown are the defaults.

```
tftp>status  
Connected to FLOWERS.COM.  
Mode: octet Tracing: off  
Rexmt-interval: 5 seconds, Max-timeout: 25 seconds  
tftp>
```

TIMEOUT

Sets the amount of time TFTP waits for a response from the server before aborting a transfer.

The REXMT command controls how often the request is retransmitted. The default value for the maximum timeout is 25 seconds.

FORMAT

timeout *seconds*

PARAMETERS

seconds

Specifies the number of seconds for the maximum timeout allowed per TFTP packet.

EXAMPLE

This example shows how to set the maximum timeout to 50 seconds.

```
tftp> timeout 50  
tftp>
```


TRACE

Toggles TFTP packet tracing.

FORMAT

trace

EXAMPLES

This example shows how to enable TFTP packet tracing. Issue the command a second time to disable packet tracing.

```
tftp>trace
Packet tracing on.
tftp>
```

This example shows a transfer with packet tracing enabled.

```
get use2s:[smith]login.com .com

sent LOCALHOST.69   RRQ <file=users:[smith]login.com, mode=octet>
received LOCALHOST.69   DATA <block=1, 512 bytes>
sent LOCALHOST.69   ACK <block=1>
received LOCALHOST.69   DATA <block=2, 512 bytes>
sent LOCALHOST.69   ACK <block=2>
received LOCALHOST.69   DATA <block=3, 512 bytes>
sent LOCALHOST.69   ACK <block=3>
received LOCALHOST.69   DATA <block=4, 512 bytes>
sent LOCALHOST.69   ACK <block=4>
received LOCALHOST.69   DATA <block=5, 313 bytes>
Received 2361 bytes in 2 seconds.
tftp>
```


Index

A

ALL-IN-1, using mail under 3-4
authentication passphrases
 over network 8-20
authentication private keys 8-20

C

CIPHER
 3des 8-6
 arcfour 8-6
 blowfish 8-6
 des 8-6
 idea 8-6
 none 8-6
cipher
 3des 8-11
 arcfour 8-11
 blowfish 8-11
 des 8-11
 idea 8-11
 none 8-11

D

DCL command
 MULTINET DECODE A-3
 MULTINET FINGER A-4
 MULTINET FTP A-5
 MULTINET KERBEROS DESTROY A-9
 MULTINET KERBEROS INIT A-10
 MULTINET KERBEROS LIST A-11
 MULTINET KERBEROS PASSWORD A-12
 MULTINET LPRM A-13
 MULTINET RCP A-14
 MULTINET REMIND A-18

MULTINET RLOGIN A-20
MULTINET RSHELL A-22
MULTINET RUSERS A-24
MULTINET SEND A-25
MULTINET TALK A-26
MULTINET TELNET A-28
MULTINET TFTP A-34
MULTINET WHOIS A-35

DECwindows, running 7-1
default cipher
 3DES 8-23

E

encrypted data 8-8

F

firewalls, transferring files from 6-15
forwarded ports
 tunnels 8-8

FTP

 anonymous 6-14
 command
 ACCOUNT B-6
 AGET B-7
 APPEND GET B-8
 APPEND PUT B-9
 APPEND RECEIVE B-10
 APPEND SEND B-11
 APUT B-12
 ASCII B-13
 ATTACH B-14
 BELL B-15
 BINARY B-16
 BLOCK B-17
 BYE B-18
 BYTE B-19
 CD B-20

CDUP B-21
CLOSE B-22
CONFIRM B-23
CONNECT B-24
CPATH B-25
CREATE-DIRECTORY B-26
CWD B-27
DELETE B-28
DIRECTORY B-29
DISCONNECT B-30
EXIT B-31
EXIT-ON-ERROR B-32
GET B-33
HASH B-34
HELP B-35
LCD B-36
LDIR B-37
LIST B-38
LOCAL-CD B-39
LOCAL-DIRECTORY B-40
LOCAL-PWD B-41
LOGIN B-42
LPWD B-43
LS B-44
MDELETE B-45
MGET B-46
MKDIR B-47
MPUT B-48
MULTIPLE DELETE B-49
MULTIPLE GET B-50
MULTIPLE PUT B-51
MULTIPLE RECEIVE B-52
MULTIPLE SEND B-53
OPEN B-54
PASSIVE B-55
PASSWORD B-57
PORT B-58
PROMPT-FOR-MISSING-ARGUMENTS
 B-59
PROMPT-ON-CONNECT B-60
PUSH B-61
PUT B-62
PWD B-64
QUIT B-65
QUOTE B-66
RECEIVE B-67
RECORD-SIZE B-68
REMOTE-HELP B-69
REMOVE-DIRECTORY B-70
RENAME B-71
RETAIN B-72
RM B-73
RMDIR B-74
SEND B-75
SET B-76
SHOW-DIRECTORY B-77

SITE B-78
SPAWN B-79
STATISTICS B-80
STATUS B-81
STREAM B-82
STRUCTURE B-83
TAKE B-84
TENEX B-85
TYPE B-86
USER B-87
VERBOSE B-88
VERSION B-89

command scripts 6-13
initialization file 6-15
log files 6-14
troubleshooting 6-16
using commands 6-5
VMS structure 6-11

H

host
 alias specifying 3-3
 equivalences 5-3
 information, displaying 2-2
HOST.EQUIV 5-4

I

IBM 3278 models 5-10
individual aliases, specifying 3-3
insecure network 8-1

K

keepalive 8-13
Kerberos
 password, changing 4-3
 understanding 4-1
keyboard mapping file format 5-13

L

logical
 DECW\$DISPLAY 8-4
 MULTINET_DISABLE_SPAWN B-14, B-61, B-79,
 C-5, C-19, C-35
 MULTINET_FTP_NONPASV 6-15
 MULTINET_FTP_WINDOW_SIZE 6-6
 MULTINET_HOST_ALIAS_FILE 3-3

MULTINET_RCP_INDEX_UPTO_EOF 6-1
MULTINET_SMTP_FROM_HOST 3-3
MULTINET_SMTP_HOST_NAME 3-3
MULTINET_TELNET_PRINT_ESCAPE_
 CHARACTER A-32
MULTINET_TN3270_APPLICATION_KEYPAD 5-19
MULTINET_TN3270_LANGUAGE 5-19, 5-20
MULTINET_TN3270_PRINTER 5-18
MULTINET_TN3270_TRANSLATION_TABLES 5-20
MULTINET_TN5250_APPLICATION_KEYPAD 5-19
MULTINET_TN5250_PRINTER 5-18
LOGIN.COM, inhibiting output from 6-3

M

MULTINET
 HOSTS.EQUIV 5-3, 8-1
 SHOST.EQUIV 8-1
MultiNet
 Secure Shell (SSH) client 8-1
MULTINET SSHADD 8-21
MULTINET SSHAGENT 8-20
MULTINET SSHKEYGEN 8-22

O

OpenVMS mail, using across the network 3-1

P

passphrase 8-20, 8-22
 forgotten 8-22
 lost 8-22
port forwarding
 definition 8-8
public-key cryptography 8-2

Q

qualifiers
 DCL
 CREATE_NTY 5-8
 ESCAPE_CHARACTER 5-6
DECwindows
 NODE 7-1
 TRANSPORT 7-1
FINGER
 CLUSTER A-4
 NOCLUSTER 2-3, A-4
FTP

ACCOUNT A-5
BINARY A-5
FDL 6-7, 6-8
IMAGE A-5
INITIALIZATION A-6
MODE A-6
NOINITIALIZATION 6-16
NONPASV 6-15
NOVMS_STRUCTURE_NEGOTIATION A-7
PASSWORD A-6
PASV 6-15
PASV DCL 6-15
PASV=NEGOTIATE 6-15
PORT A-6
PROMPT A-6
STATISTICS A-6
STRUCTURE A-6
TAKE_FILE A-7
TYPE A-7
TYPE=EBCDIC 6-8
USERNAME A-7
VERBOSE A-7
VMS_STRUCTURE_NEGOTIATION A-7
WINDOW_SIZE 6-6

GET

FDL B-33

KERBEROS

AUTH 4-2, 4-3
AUTHENTICATION=KERBEROS 4-3
CHECK_TGT 4-3
REALM 4-2
USERNAME 4-2, 4-3

KERBEROS DESTROY

QUIET A-9
STATUS A-9

KERBEROS INIT

INSTANCE A-10
LIFETIME A-10
REALM A-10
USERNAME A-10
VERBOSE A-10

KERBEROS LIST

BRIEF A-11
CHECK_TGT A-11
SRVTAB A-11

KERBEROS PASSWORD

INSTANCE A-12
REALM A-12
USERNAME A-12

LPRM

ALL A-13
NODE A-13
QUEUE A-13
SUPERUSER A-13
USER A-13

PUT

- RCP
 - FDL B-62
 - AUTHENTICATION=KERBEROS A-14
 - EXACT A-15
 - LOG A-15
 - PASSWORD 6-2, A-15
 - RECURSIVE A-14, A-15
 - TRUNCATE_USERNAME A-15
 - USERNAME 6-2, A-15
 - VMS_ATTRIBUTES A-16
 - RECEIVE
 - FDL B-67
 - RLOGIN
 - AUTHENTICATION=KERBEROS A-20
 - BUFFER_SIZE A-20
 - DEBUG A-20
 - EIGHT_BIT A-20
 - PORT A-20
 - TRUNCATE_USERNAME A-20
 - USERNAME A-21
 - RSHELL
 - ERROR 5-2, A-22
 - INPUT 5-2, A-22
 - INPUT=NLA0
 - 5-2
 - OUTPUT 5-2, A-22
 - PASSWORD 5-2, A-22, A-23
 - PORT A-22
 - TRUNCATE_USERNAME A-23
 - USERNAME 5-2, A-23
 - RUSERS
 - ALL A-24
 - FULL A-24
 - NOALL A-24
 - NOFULL A-24
 - SEND
 - AND_MAIL A-25
 - FDL B-75
 - FOREIGN A-3
 - OR_MAIL A-25
 - SET
 - ABORT_OUTPUT_CHARACTER C-21
 - ACCOUNT B-76
 - ARE_YOU_THERE_CHARACTER C-22
 - AUTH C-33
 - AUTOFLUSH C-23
 - BREAK_CHARACTER C-24
 - DEBUG C-25
 - ERASE_CHARACTER_CHARACTER C-26
 - ERASE_LINE_CHARACTER C-27
 - ESCAPE_CHARACTER C-28
 - INTERRUPT_PROCESS_CHARACTER
 - C-30
 - LOCAL_FLOW_CONTROL C-31
 - LOG_FILE C-32
 - PASSWORD B-76
 - UNIX C-34
 - USER B-76
 - SPAWN
 - INPUT B-79, C-35
 - LOGICAL_NAMES B-79, C-35
 - OUTPUT B-79, C-35
 - SYMBOLS B-79, C-35
 - WAIT B-79, C-35
 - TALK
 - OLD A-26
 - TELNET
 - ABORT_OUTPUT_CHARACTER A-28
 - ARE_YOU_THERE_CHARACTER A-28
 - AUTHENTICATION=KERBEROS A-28
 - AUTOFLUSH A-28
 - BREAK_CHARACTER A-28
 - BUFFER_SIZE A-29
 - CREATE_NTY A-29
 - DEBUG A-30
 - DELETE_NTY A-30
 - ERASE_CHARACTER_CHARACTER A-30
 - ERASE_LINE_CHARACTER A-30
 - ESCAPE_CHARACTER A-30
 - INTERRUPT_PROCESS_CHARACTER
 - A-31
 - LOCAL_FLOW_CONTROL A-31
 - LOG_FILE A-31
 - PORT A-31
 - PRINT_ESCAPE_CHARACTER A-32
 - PROTOCOL A-32
 - TCP A-32
 - TERMINAL_TYPE A-32, C-37
 - TN3270=AUTOMATIC A-32
 - TN5250 5-9
 - TN5250=AUTOMATIC A-32
 - UNIX A-32
 - VERSION A-33
 - TN3270
 - YALE 5-19
 - WHOIS
 - HOST A-35
 - OUTPUT A-35
 - PORT A-35
-
- R**
- R services
 - authentication 5-3
 - RCP
 - requirements 6-1
 - using 6-1, 6-2
 - using Kerberos with 4-3
 - remote hosts, specifying 2-1
 - remote login program
 - first authentication method 8-1

- fourth authentication method 8-4
- second authentication method 8-1
- third authentication method 8-2
- RHOSTS 5-4
- RhostsAuthentication 8-14
- RhostsRSAAuthentication 8-14
- RLOGIN
 - terminating 5-3
 - using 5-2
 - using Kerberos with 4-3
- RSA authentication 8-20
- RSA authentication identity 8-22, 8-24
- RSA-based authentication 8-2
- RSA-based host authentication 8-1
- RSHELL
 - executing commands 5-1
 - using 5-1
 - using Kerberos with 4-3

S

- secure shell
 - configuration file
 - keyword
 - BatchMode 8-11
 - Cipher 8-11
 - ClearAllForwardings 8-11
 - Compression 8-11
 - CompressionLevel 8-11
 - ConnectionAttempts 8-11
 - EscapeChar 8-11
 - FallBackToRsh 8-12
 - ForwardAgent 8-12
 - ForwardX11 8-12
 - GatewayPorts 8-12
 - GlobalKnownHostsFile 8-12
 - Host 8-12
 - IdentityFile 8-12
 - KeepAlive 8-13
 - LocalForward 8-13
 - NumberOfPasswordPrompts 8-13
 - PasswordAuthentication 8-13
 - PasswordPromptHost 8-13
 - PasswordPromptLogins 8-13
 - Port 8-13
 - ProxyCommand 8-14
 - RemoteForward 8-14
 - RhostsAuthentication 8-14
 - RhostsRSAAuthentication 8-14
 - RSAAuthentication 8-14
 - StrictHostKeyChecking 8-15
 - UsePrivilegedPort 8-15
 - UserKnownHostsFile 8-15
 - UseRsh 8-15
 - configuration files 8-10
 - secure shell client 8-1
 - spoofing
 - DNS 8-1
 - IP 8-1
 - routing 8-1
 - SSH
 - authentication agent 8-20
 - command options 8-6
 - SSH command
 - ALLOW_REMOTE_CONNECT 8-6
 - CIPHER 8-6
 - COMPRESSION 8-6
 - DEBUG 8-6
 - ESCAPE_CHARACTER 8-7
 - IDENTITY_FILE 8-7
 - LOCAL_FORWARD 8-7
 - LOG_FILE 8-7
 - NO_AGENT_FORWARDING 8-7
 - OPTION 8-7
 - PORT 8-7
 - QUIET 8-8
 - REMOTE_FORWARD 8-8
 - USE_NONPRIV_PORT 8-8
 - USERNAME 8-8
 - VERSION 8-8
 - SSH files
 - AUTHORIZED_KEYS 8-16
 - CONFIG 8-16
 - HOSTS.EQUIV 8-18
 - IDENTITY 8-16
 - IDENTITY.PUB 8-16
 - KNOWN_HOSTS 8-17
 - RANDOM_SEED 8-17
 - RHOSTS 8-18
 - SHOSTS 8-18
 - SHOSTS.EQUIV 8-19
 - SSH_CONFIG 8-19
 - SSH_KNOWN_HOSTS 8-19
 - SSHADD 8-20, 8-21
 - SSHADD option
 - DELETE 8-21
 - LIST 8-21
 - PURGE 8-21
 - SSHAGENT 8-20
 - authentication agent 8-21
 - authentication private keys 8-20
 - SSHKEYGEN 8-22
 - authentication key pairing 8-22
 - definition 8-22
 - file
 - IDENTITY 8-24
 - IDENTITY.PUB 8-24
 - RANDOM_SEED 8-24
 - option
 - BITS 8-23

CHANGE_CIPHER 8-23
CHANGE_COMMENT 8-23
CHANGE_PASSPHRASE 8-23
COMMENT 8-23
HOST 8-23
IDENTITY_FILE 8-23
NEW_PASSPHRASE 8-23
PASSPHRASE 8-23

SYLOGIN.COM, inhibiting output from 6-3
SYS\$LOGIN
.RHOSTS 5-3

T

TELNET

command

ABORT C-4
ATTACH C-5
ATTN C-6
AYT C-7
BINARY C-8
BREAK C-9
BYE C-10
CLOSE C-11
CONNECT C-12
CREATE-NTY C-13
DEBUG C-14
ECHO C-15
EXIT C-16
HELP C-17
LOG-FILE C-18
PUSH C-19
QUIT C-20
SET ABORT-OUTPUT-CHARACTER C-21
SET ARE-YOU-THERE-CHARACTER C-22
SET AUTO-FLUSH C-23
SET BREAK-CHARACTER C-24
SET DEBUG C-25
SET ERASE-CHARACTER-
CHARACTER C-26
SET ERASE-LINE-CHARACTER C-27
SET ESCAPE-CHARACTER C-28
SET EXTENDED C-29
SET INTERRUPT-PROCESS-
CHARACTER C-30
SET LOCAL-FLOW-CONTROL C-31
SET LOG-FILE C-32
SET REMOTE-USERNAME C-33
SET UNIX-LINE-TERMINATOR C-34
SPAWN C-35
STATUS C-36
TERMINAL-TYPE C-37
VERSION C-38

commands, using 5-5

control sequence
ABORT-OUTPUT 5-7
ARE-YOU-THERE 5-7
BREAK-CHARACTER 5-7
ERASE-CHARACTER 5-8
ERASE-LINE 5-8
INTERRUPT-PROCESS 5-8

control sequences, using 5-7
logging in with 5-5
starting 5-5
troubleshooting 5-21
using Kerberos with 4-3

TELNET sessions 8-8

TFTP

command

CONNECT D-2
GET D-3
PUT D-4
QUIT D-5
REXMT D-6
STATUS D-7
TIMEOUT D-8
TRACE D-9

copying files using 6-17
requirements 6-17
using 6-17

ticket status, checking 4-3

tickets, acquiring and deleting 4-2

TN3270

application keypad access 5-19
emulation 5-19
function key mapping 5-14
translation table mapping 5-19
using transparent mode 5-18

TN5250

application keypad access 5-19
TN5250 function key mapping 5-16

tunneling 8-8

typographical conventions 1-2

U

unsecure connections 8-8

untrusted hosts 8-1

user

equivalences 5-3
information, displaying 2-2

utility

PHONE 2-4
RLOGIN 5-1
RSHELL 5-1
TALK 2-4
TELNET 5-1

W

WHOIS A-35

X

Xauthority data 8-5

MultiNet Master Index

Guide to Abbreviations

AD—*MultiNet Administrator's Guide*

AR—*MultiNet Administrator's Reference*

DN—*MultiNet TCP/IP Services for DECnet Applications*

IN—*MultiNet Installation and Introduction*

ME—*MultiNet Messages and Logicals*

PR—*MultiNet Programmer's Reference*

UG—*MultiNet User's Guide*

A

abandoned leases AD 12-66
accept() PR 2-4
access to the XDM server AD 13-8
ACCESS-CONFIG
 command
 ADD AR 8-4
 ATTACH AR 8-5
 EXIT AR 8-6
 GET AR 8-7
 HELP AR 8-8
 NETCONTROL AR 8-9
 PUSH AR 8-10
 QUIT AR 8-11
 RELOAD AR 8-12
 REMOVE AR 8-13
 SAVE AR 8-14
 SET AR 8-15
 SHOW AR 8-19
 SPAWN AR 8-20
 STATUS AR 8-22

USE AR 8-23
VERSION AR 8-24
WRITE AR 8-25
 using AD 4-26
ACL
 support over NFS AD 19-24
 with unmappable ACEs AD 19-27
ACPs (Ancillary Control Processes) AD 20-14
add or update user exits IN 1-22
address
 lease states in DHCP dump files AD 12-80
 pools for all subnets AD 12-77
 pools for specific subnets AD 12-77
address_match_list AD 6-24
advanced XDM resources AD 13-6
ADVISORY_CLOSE AD 20-16
AF_CHAOS PR 2-1
AF_INET PR 2-1, PR 2-2, PR 2-6
AgentX peers AD 15-9
ALL-IN-1, using mail under UG 3-4
Apple Macintosh users IN 1-19
ARP (Address Resolution Protocol) IN 6-13
 table AD 3-51
AST reentrancy PR 3-1
auth AD 7-15
authentication
 concepts AD 3-27
 systems, managing AD 3-32
 within a trusted local network AD 3-29
authentication agent connection AD 21-2
authentication passphrases
 over network UG 8-20
authentication private keys UG 8-20
AUTHORIZED_KEYS AD 21-13
auto server cache sizing AD 19-43
available fonts AD 14-5

B

Bellcore S/KEY "Soft Token" AD 3-31

- BGP protocol
 - configuring AD 5-11
- BIND AD 21-8
 - 8.2.3 AD 6-6
- bind() PR 2-4, PR 2-5
- BOOTP AD 12-4
 - (Bootstrap Protocol) AD 12-2
 - “to” option values AD 12-9
 - clients, obtaining data for AD 12-5
 - configuration file guidelines AD 12-9
 - OPCOM messages AD 12-10
 - options AD 12-6
- broadcast AD 7-14
- broadcastclient AD 7-15
- BSD PR 2-5

C

- cache
 - interrupt parameters AD 19-40
 - maintenance interval parameters AD 19-41
 - memory requirements parameters AD 19-43
 - refresh parameters AD 19-41
 - size parameters AD 19-42
 - timing parameters AD 19-41
- caching-only name server AD 6-9
- channel deassignment parameters AD 19-41
- channels, file headers, and data buffers AD 19-39
- CHECK IN 6-3
- CIPHER
 - 3des UG 8-6
 - arcfour UG 8-6
 - blowfish UG 8-6
 - des UG 8-6
 - idea UG 8-6
 - none UG 8-6
- cipher
 - 3DES AD 21-2
 - 3des UG 8-11
 - ARCFOUR AD 21-2
 - arcfour UG 8-11
 - BLOWFISH AD 21-2
 - blowfish UG 8-11
 - DES AD 21-2
 - des UG 8-11
 - IDEA AD 21-2
 - idea UG 8-11
 - none UG 8-11
- client classes AD 12-21
- cluster service
 - names, monitoring AD 6-36
 - setting up a AD 6-35
- cluster-wide aliases AD 19-17
- command-line interface utilities AD 2-3

- compiled-in timezone rules AD 7-2
- concurrency parameters AD 19-40
- conditional behavior AD 12-24
- configuration
 - recommendations AD 3-25
 - tasks AD 2-1
 - configuring services AD 2-2
 - establishing basic IP connectivity AD 2-1
 - utilities AD 2-2
- configure global parameters AD 6-35
- connect() PR 2-3, PR 2-5, PR 4-1
- controlkey AD 7-17
- CONVNTP AD 7-12
- CRASH-ON-EXCEPTION AD 19-46
- cross-realm authentication AD 16-13
- CRYPTOCARD
 - /DISPLAY keyword
 - DECIMAL AR 2-10
 - HEXADECIMAL AR 2-11
 - TELEPHONE AR 2-11
 - USERID AR 2-11
 - /KEY keyword
 - NUMBER AR 2-11
 - OCTAL AR 2-11
 - SPLIT AR 2-11
 - VALUE AR 2-12
 - /PIN parameter
 - {FEEDBACK | NOFEEDBACK} AR 2-13
 - FIXED AR 2-13
 - LENGTH AR 2-13
 - TRIES AR 2-13
- authentication AD 4-35
- token AD 3-30

D

- d2 AD 6-33
- dassgn AD 4-16
- databases DN 1-2
- DCL command
 - MULTINET DECODE UG A-3
 - MULTINET FINGER UG A-4
 - MULTINET FTP UG A-5
 - MULTINET KERBEROS DESTROY UG A-9
 - MULTINET KERBEROS INIT UG A-10
 - MULTINET KERBEROS LIST UG A-11
 - MULTINET KERBEROS PASSWORD UG A-12
 - MULTINET LPRM UG A-13
 - MULTINET RCP UG A-14
 - MULTINET REMIND UG A-18
 - MULTINET RLOGIN UG A-20
 - MULTINET RSHELL UG A-22
 - MULTINET RUSERS UG A-24
 - MULTINET SEND UG A-25
 - MULTINET TALK UG A-26

- MULTINET TELNET UG A-28
- MULTINET TFTP UG A-34
- MULTINET WHOIS UG A-35
- DCLTABLES.EXE file IN 1-8
 - installing MultiNet commands IN 1-23
- debug AD 6-33
- DEBUG-MESSAGE-CACHE-SIZE AD 19-46
- DECnet
 - application services DN 1-1
 - configuring DN 2-1
 - considerations DN 1-2
 - starting without rebooting DN 2-3
 - application services, testing DN 2-4
 - client
 - access to an IP server AD 17-4
 - on the IP server AD 17-6
 - encapsulation over unreliable networks AD 18-3
 - networking management DN 1-3
 - over-IP circuits AD 18-1
 - to-SMTP mail AD 8-32
- DECNET-CONFIG command
 - ADD AR 3-3
 - ATTACH AR 3-4
 - CLEAR AR 3-6
 - DELETE AR 3-7
 - ERASE AR 3-8
 - EXIT AR 3-9
 - GET AR 3-10
 - HELP AR 3-11
 - MODIFY AR 3-12
 - PUSH AR 3-13
 - QUIT AR 3-14
 - SAVE AR 3-15
 - SHOW AR 3-16
 - SPAWN AR 3-17
 - STATUS AR 3-19
 - USE AR 3-20
 - VERSION AR 3-21
 - WRITE AR 3-22
- DECstation
 - mount points AD 19-32
 - systems AD 19-31
- DECwindows, running UG 7-1
- default cipher
 - 3DES UG 8-23
- default LPD user name AD 9-3
- defname AD 6-33
- DELETE AR 10-9
- delete-behind cache parameters AD 19-45
- DHCP
 - (Dynamic Host Configuration Protocol) AD 12-2
 - address
 - allocation AD 12-19
 - pools AD 12-20
 - lease state
 - abandoned AD 12-80
 - bound AD 12-80
 - free AD 12-80
 - offered AD 12-80
 - pinging AD 12-80
 - reserved for secondary AD 12-80
 - static assignment AD 12-81
 - pools AD 12-20
- agent option space options AD 12-61
- clients AD 12-13
- configuration AD 12-14
 - file statement
 - backup-ack-interval AD 12-71
 - backup-pool-size AD 12-71
 - com-int-timeout AD 12-72
 - failover-port AD 12-72
 - mclt AD 12-72
 - safe-period-timeout AD 12-72
 - startup-delay AD 12-72
- conversion tool AD 12-15
- declaration AD 12-17
 - group AD 12-17
 - host AD 12-17, AD 12-19
 - pool AD 12-20
 - range AD 12-17
 - shared-network AD 12-17
 - subnet AD 12-17
- failover protocol AD 12-72
- lease
 - file statement
 - abandoned AD 12-65
 - billing class AD 12-65
 - billing subclass AD 12-65
 - client-hostname AD 12-65
 - domain-name AD 12-65
 - dynamic-bootp AD 12-65
 - ends AD 12-65
 - FQDN AD 12-65
 - hardware AD 12-65
 - hostname AD 12-66
 - starts AD 12-66
 - uid AD 12-66
 - format AD 12-64
- leases AD 12-66
- option space option
 - option all-subnets-local AD 12-53
 - option arp-cache-timeout AD 12-53
 - option bootfile-name AD 12-53
 - option boot-size AD 12-53
 - option broadcast-address AD 12-53
 - option cookie-servers AD 12-53
 - option default-ip-ttl AD 12-53
 - option default-tcp-ttl AD 12-53
 - option dhcp-client-identifier AD 12-53
 - option dhcp-max-message-size AD 12-54
 - option dhcp-parameter-request-list AD 12-54
 - option dhcp-server-identifier AD 12-54

- option domain-name AD 12-54
- option domain-name-servers AD 12-54
- option extensions-path AD 12-55
- option finger-server AD 12-55
- option font-servers AD 12-55
- option host-name AD 12-55
- option ieee802-3-encapsulation AD 12-55
- option ien116-name-servers AD 12-55
- option impress-servers AD 12-55
- option interface-mtu AD 12-55
- option ip-forwarding AD 12-55
- option irc-server AD 12-55
- option log-servers AD 12-55
- option lpr-servers AD 12-56
- option mask-supplier AD 12-56
- option max-dgram-reassembly AD 12-56
- option merit-dump AD 12-56
- option mobile-ip-home-agent AD 12-56
- option nds-context AD 12-56
- option nds-servers AD 12-56
- option nds-tree-name AD 12-56
- option netbios-dd-server AD 12-56
- option netbios-name-servers AD 12-56
- option netbios-node-type AD 12-56
- option netbios-scope AD 12-57
- option nis-domain AD 12-57
- option nisplus-domain AD 12-57
- option nisplus-servers AD 12-57
- option nis-servers AD 12-57
- option nntp-server AD 12-57
- option non-local-source-routing AD 12-57
- option ntp-servers AD 12-57
- option option AD 12-57
- option path-mtu-aging-timeout AD 12-57
- option path-mtu-plateau-table AD 12-57
- option perform-mask-discovery AD 12-58
- option policy-filter AD 12-58
- option pop-server AD 12-58
- option resource-location-servers AD 12-58
- option root-path AD 12-58
- option router-discovery AD 12-58
- option routers AD 12-58
- option router-solicitation-address AD 12-58
- option smtp-server AD 12-58
- option static-routes AD 12-59
- option streettalk-directory-assistance-server AD 12-59
- option streettalk-server AD 12-59
- option subnet-mask AD 12-59
- option swap-server AD 12-59
- option tcp-keepalive-garbage AD 12-59
- option tcp-keepalive-interval AD 12-59
- option tftp-server-name AD 12-60
- option time-offset AD 12-60
- option time-servers AD 12-60
- option trailer-encapsulation AD 12-60

- option vendor-encapsulated-options AD 12-60
- option www-server AD 12-60
- option x-display-manager AD 12-60
- option type
 - ARRAYS AD 12-62
 - BOOLEAN AD 12-61
 - DATA STRING AD 12-62
 - INTEGER AD 12-62
 - IP-ADDRESS AD 12-62
 - RECORDS AD 12-63
 - TEXT AD 12-62
- options AD 12-16, AD 12-52
- process AD 12-11
- relay agent information option AD 12-60
- Safe-failover AD 12-67, AD 12-68
 - lease file statement
 - acked-sec-interval AD 12-72
 - acked-sec-interval-start AD 12-72
 - active AD 12-73
 - backup AD 12-73
 - desired-interval AD 12-73
 - expired AD 12-73
 - free AD 12-73
 - last-partner-transaction AD 12-73
 - released AD 12-73
 - reset AD 12-73
 - revoked AD 12-73
 - safe-lease AD 12-73
 - transaction-id AD 12-73
 - update-count AD 12-73
 - partner down state AD 12-73
 - server mode
 - primary AD 12-70
 - secondary AD 12-70
 - standalone AD 12-70
 - server state
 - backup-comint AD 12-71
 - backup-conflict AD 12-71
 - backup-normal AD 12-71
 - backup-partnerdown AD 12-71
 - backup-recover AD 12-71
 - failover-disabled AD 12-71
 - primary-comint AD 12-71
 - primary-conflict AD 12-71
 - primary-normal AD 12-71
 - primary-partnerdown AD 12-71
 - primary-recover AD 12-71
 - startup AD 12-71
 - state file AD 12-70
- server parameter
 - ACCOUNTING AD 12-74
 - CONFIGFILE AD 12-74
 - DEBUG AD 12-74
 - DEBUG-FILE AD 12-74
 - DUMPFIL AD 12-74

- IMAGE-NAME AD 12-74
- LEASEFILE AD 12-75
- LOG-DATE AD 12-75
- LOG-TO-OPCOM AD 12-75
- PROCESS-NAME AD 12-75
- SWAP AD 12-75
- SYS-ERROR AD 12-75
- SYS-OUTPUT AD 12-75
- spawning class AD 12-24
- statement
 - add AD 12-28
 - allow and deny AD 12-28
 - allow and deny in pool declarations AD 12-31
 - allow and deny in scope AD 12-29, AD 12-30
 - always-broadcast AD 12-31
 - always-reply-rfc1048 AD 12-32
 - authoritative AD 12-33
 - class AD 12-34
 - default-lease-time AD 12-34
 - dynamic-bootp-lease-cutoff AD 12-35
 - dynamic-bootp-lease-length AD 12-35
 - filename AD 12-35
 - fixed-address AD 12-36
 - get-lease-hostnames AD 12-36
 - group AD 12-36
 - hardware AD 12-37
 - host AD 12-38
 - if AD 12-39
 - invalid-ddns-chars AD 12-39
 - lease limit AD 12-39
 - lease-scan-interval AD 12-39
 - match AD 12-40
 - match if AD 12-40
 - max-delayed-acks AD 12-40
 - max-lease-time AD 12-40
 - min-lease-time AD 12-40
 - min-secs AD 12-41
 - next-server AD 12-41
 - one-lease-per-client AD 12-41
 - option AD 12-42
 - option definition AD 12-42
 - option space AD 12-42
 - parameter AD 12-16
 - ping AD 12-42
 - ping-retries AD 12-42
 - ping-timeout AD 12-42
 - pool AD 12-43
 - range AD 12-43
 - requested-options-only flag AD 12-44
 - server-identifier AD 12-44
 - server-name AD 12-44
 - shared-network AD 12-16, AD 12-45
 - site-option-space AD 12-45
 - spawn with AD 12-45
 - subnet AD 12-16, AD 12-46
 - use-host-decl-names AD 12-47
 - use-lease-addr-for-default-route AD 12-47
 - user-class AD 12-47
 - vendor-class AD 12-47
 - vendor-option-space AD 12-48
- subclass AD 12-22
- using AD 12-11
- DHCPD.CONF file AD 12-81
- DIG, using to debug DNS AD 6-33
- directory and file
 - cache parameters AD 19-39
 - times AD 19-39
- DISKQUOTA limitations AD 20-4
- DNA and TCP/IP protocols DN 1-2
- DNS
 - (Domain Name System) IN 6-11
 - domains IN 6-11
 - dynamic updates within DHCP AD 12-25
 - host tables IN 6-11, IN 6-12
 - incremental zone transfer AD 6-28
 - load balancing AD 6-34
 - logging option
 - category AD 6-26
 - channel AD 6-26
 - file AD 6-26
 - print-category AD 6-26
 - print-severity AD 6-26
 - print-time AD 6-26
 - severity AD 6-26
 - syslog daemon AD 6-26
- NAMED.CONF option
 - allow-query AD 6-18
 - allow-recursion AD 6-19
 - allow-transfer AD 6-19
 - also-notify AD 6-19
 - blackhole AD 6-19
 - check-names AD 6-20
 - directory AD 6-20
 - fake-iquery AD 6-20
 - fetch-glue AD 6-20
 - forward AD 6-21
 - forwarders AD 6-21
 - listen-on AD 6-22
 - maintain-ixfr-base AD 6-22
 - max-ixfr-log-size AD 6-22
 - min-roots AD 6-22
 - notify AD 6-22
 - recursion AD 6-22
 - rrset-order AD 6-22
 - sortlist AD 6-23
 - topology AD 6-23
 - transfer-source AD 6-23
 - version AD 6-23
- NSLOOKUP command
 - exit AD 6-33
 - finger AD 6-33
 - help AD 6-33

- ls AD 6-34
 - lserver AD 6-34
 - name AD 6-33
 - name server AD 6-33
 - root AD 6-34
 - server AD 6-34
 - set all AD 6-33
 - set class AD 6-33
 - set domain AD 6-33
 - set port AD 6-33
 - set query-type AD 6-34
 - set retry AD 6-33
 - set root AD 6-33
 - set srchlist AD 6-33
 - set timeout AD 6-33
 - set type AD 6-34
 - optional zone statement
 - allow-query AD 6-15
 - allow-transfer AD 6-15
 - allow-update AD 6-15
 - also-notify AD 6-16
 - check-names AD 6-16
 - forward AD 6-16
 - forwarders AD 6-16
 - hint AD 6-17
 - ixfr-base AD 6-16
 - master AD 6-17
 - notify AD 6-16
 - pubkey AD 6-16
 - transfer-source AD 6-16
 - resolver (client) AD 6-8
 - resolver routines PR 3-10
 - resolvers and servers AD 6-8
 - resource record sorting AD 6-26
 - resource record type
 - A AD 6-29
 - CNAME AD 6-29
 - HINFO AD 6-29
 - KEY AD 6-29
 - MB AD 6-29
 - MG AD 6-29
 - MINFO AD 6-29
 - MR AD 6-29
 - MX AD 6-29
 - NS AD 6-29
 - NULL AD 6-29
 - NXT AD 6-29
 - PTR AD 6-30
 - SIG AD 6-30
 - SOA AD 6-30
 - SRV AD 6-30
 - TXT AD 6-30
 - WKS AD 6-30
 - security AD 6-37
 - server IN 6-11, AD 6-9
 - using DN 2-6, AD 6-8
 - zone information files AD 6-28
 - zone type
 - forward AD 6-17
 - slave AD 6-18
 - stub AD 6-18
 - DNS and BIND, Third Edition AD 6-8
 - documentation
 - comments AD 1-8
 - online IN 1-23
 - structure PR 1-1
 - domain name versus host name AD 6-13
 - DOMAIN-NAME-SERVICE.CACHE AD 6-17
 - duplicate request detection cache parameters AD 19-45
 - dynamic configuration protocol IN 6-8
 - BOOTP (Bootstrap Protocol) IN 6-9
 - DHCP (Dynamic Host Configuration Protocol) IN 6-9
 - RARP (Reverse Address Resolution Protocol) IN 6-9
-
- ## E
- EGP protocol
 - configuring AD 5-10
 - electronic mail AD 1-5
 - empty passwords AD 21-9
 - encrypted data UG 8-8
 - errno PR 4-1
 - ERRNO.H error values AD 17-7
 - error message
 - %BACKUP-E-OPENOUT ME 2-4
 - %CLI-F-SYNTAX ME 2-11
 - %COPY-E-OPENOUT ME 2-15
 - %CREATE-E-
 - OPENOUT ME 2-18
 - READERR ME 2-19
 - %DCL-W-ACTIMAGE ME 2-20, ME 2-21
 - %DECW
 - F-CANT_OPEN-DISPL ME 2-21
 - I-ATTACHED ME 2-22
 - W-ATT_FAIL ME 2-22
 - %DIRECT-E-OPENIN ME 2-23
 - %DISM-W-CANNOTDMT, NFSn ME 2-24
 - %JBC-E-SYMDEL ME 2-37
 - %LIB-E-ACTIMAGE ME 2-45
 - %LINK-F-NOLINKSTB ME 2-44
 - %MAIL-E-
 - ERRACTRNS ME 2-45
 - OPENIN ME 2-46
 - USERSPEC ME 2-46
 - %MAIL-E-ERRACTRNS ME 2-45
 - %MNCHK-I-LOGVRFY ME 2-48
 - %MNCHK-W-
 - DBVERMISM ME 2-48
 - DUPPHA, LAN ME 2-49
 - NOROOTNS ME 2-49

- %MOUNT-F-
 - CTRLERR ME 2-50
 - UNSAFE ME 2-51
 - VOLALRMNT ME 2-51
- %MULTINET-E-BADVMS ME 2-52
- %MULTINET-F-
 - EACCES ME 2-102
 - EAFNOSUPPORT ME 2-52
 - ECONNCLOSED ME 2-53
 - ECONNREFUSED ME 2-8
 - EHOSTUNREACH ME 2-54
 - ERREADFLAGS ME 2-54
 - ETIMEDOUT ME 2-55
 - MTUERR ME 2-55
 - SETPORT ME 2-56
- %MULTINET-W-ENOBUFFS ME 2-54
- %NFSDISMNT-F-DISMOUNTERR ME 2-68
- %NFSMOUNT-F-
 - BADRPCTRANSPORT ME 2-68, ME 2-69
 - BADRSIZE ME 2-69
 - BADTIMEOUT ME 2-70
 - BADWSIZE ME 2-70
 - CANTLOAD ME 2-70
 - MOUNTERR ME 2-70
 - NFSERR ME 2-71
 - NOACCESS ME 2-71
 - NOMOUNTRESPONSE ME 2-71
 - NONFSRESPONSE ME 2-71
 - NOREMOTEHOST ME 2-72
 - RPCCREATEERROR ME 2-69
- %NFSMOUNT-I-
 - MOUNTED ME 2-72
 - WAITDNS ME 2-72
- %NFSMOUNT-W-
 - NOPRIVMOUNTPORT ME 2-72
 - NOPRIVNFSPORT ME 2-73
- %NTYCP-E-
 - CREATERR ME 2-78
 - DELETERR ME 2-78
 - DEVNAMERR ME 2-77
 - NOADDR ME 2-77
 - NOSUCHNODE ME 2-77
 - NOSUCHPORT ME 2-77
 - NOTINET ME 2-77
 - NOTNTY ME 2-79
 - PORTRNG ME 2-78
 - PORTSYNTAX ME 2-78
- %NTYCP-F-
 - CMDINITERR ME 2-79
 - INTERNAL_ERROR ME 2-79
- %NTYCP-S-
 - CREPORT ME 2-79
 - DELPOR ME 2-79
- %NTYCP-W-
 - CMDERR ME 2-80
 - LOGDEFERR ME 2-80
- NODEREQD ME 2-80
- OPENCMD ME 2-80
- PORTREQD ME 2-80
- %PATCH-I-
 - NOGBL ME 2-84
 - NOLCL ME 2-84
- %PSM-E-
 - OPENIN error opening
 - filename as input ME 2-86
 - SY\$LIBRARY ME 2-86
 - READERR, error reading
 - !AS -RMS-W-RTB, !UL ME 2-87
 - print_file -RMS-W-RTB ME 2-87
 - WRITEERR, error writing
 - !AS -SYSTEM-F-NOTPRINTED ME 2-87
 - !AS -SYSTEM-W-NOMSG ME 2-87
- %QMAN-I-INVSMBSMSG ME 2-102
- %QUEMAN-F-OPENOUT ME 2-88
- %RCP-F-ERROR, message ME 2-88
- %RMS-
 - E-DNF, directory not found ME 2-89
 - W-RTB, byte record too large for user's buffer ME 2-89
- %RMT-F-ALLOCERR, Error allocating RCD0 device ME 2-89
- %SET-E-
 - DEVOFFLINE ME 2-93
 - NOMSG ME 2-93
- %SET-W-NOTSET ME 2-94
- %SYSMAN-I-NODERR ME 2-96
- %SYSTEM-E-NORIGHTSDB ME 2-96
- %SYSTEM-F-
 - ACCVIO ME 2-96
 - reason mask=00 ME 2-97
 - reason mask=04 ME 2-97
 - ILLIOFUNC ME 2-99
 - INTDIV ME 2-99
 - NOLOGNAM ME 2-99
- %TPU-E-OPENOUT ME 2-102
- %UCX-E-LPD_REQREJECT ME 2-102
- %X11DEBUG
 - F-REFUSED, X11 ME 2-107
 - F-UNKNOWNHOST ME 2-108
- %X11DEBUG-F-
 - CONNECTFAIL ME 2-107
- absurdly long client literal user=user ME 2-1
- accept_deenet
 - \$ASSIGN error ME 2-1
 - \$TRNLNM error ME 2-1
- accept_tcp
 - \$ASSIGN error ME 2-2
 - Error! Server port number is below 6000 ME 2-2
 - getpeername error ME 2-2
 - getsockname error ME 2-2
 - setsockopt error ME 2-2

- access denied ME 2-3
- access_file I/O error, VMS Status ME 2-3
- append to folder failed ME 2-3
- authenticated user ME 2-3
- auto-filing of INBOX messages not completed ME 2-3
- bad
 - address (EFAULT) ME 2-4
 - address-address ME 2-4
 - instance ME 2-5
 - name ME 2-5
 - realm ME 2-5
- BRTOOFAR ME 2-5
- buffer
 - didn't grow ME 2-6
 - extend failed ME 2-6
 - extend failed in read ME 2-6
 - incorrectly in bitmap ME 2-7
 - n incorrectly in bitmap ME 2-7
- buffer_flush I/O error, VMS Status ME 2-6
- buffer_read I/O error, VMS Status ME 2-7
- BYE host Fatal mailbox error ME 2-7
- byte order is value ME 2-7
- can't
 - build data connection ME 2-8
 - create subscription database ME 2-8
 - create subscription temporary file ME 2-8
 - open database ME 2-8
 - resize free storage ME 2-9
 - write subscription temporary file ME 2-9
- cannot
 - connect to MR ME 2-9
 - rename new empty folder list ME 2-10
 - save old empty folder list ME 2-10
 - update folder database ME 2-10
 - write sequential mail ME 2-10
- CLI-E-IMGNAME, image file filename ME 2-11
- client
 - NFS unknown attribute ME 2-11
 - not authorized to access server ME 2-11
- CNXMAN, lost connection to host node ME 2-12
- command
 - not understood ME 2-12
 - stream end of file while reading char user ME 2-12
 - line user ME 2-12
- compiler bugcheck ME 2-12
- complete_decnets
 - Inbound \$QIO IOSB ME 2-13
 - Outbound socket error ME 2-13
- complete_tcp
 - %QIOW Outbound Error ME 2-14
 - Inbound \$QIO IOSB ME 2-13
 - Inbound End Of File ME 2-13
 - QIOW IOSB Outbound Error ME 2-14
- connection refused ME 2-14
- corrupted setup file ME 2-15
- could not
 - open alternate database name ME 2-15
 - remove old empty folder list ME 2-15
- couldn't
 - bind to control socket ME 2-16
 - create database multinet
 - kerberos_principal ME 2-16
 - create file ME 2-16
 - create temp database temp_file ME 2-17
 - get master key ME 2-17
 - read master key ME 2-17
 - store principal.instance ME 2-17
 - translate VMS error ME 2-18
 - translate VMS error 828 ME 2-18
- create_file
 - I/O error, VMS Status ME 2-19
 - write error, VMS Status ME 2-19
- cross-device link (EXDEV) ME 2-19
- database rename failed ME 2-19
- date invalid ME 2-20
- DCL-W-ACTIMAGE ME 2-20
- deaccess I/O error, VMS Status ME 2-21
- DECnet connection is from server ME 2-21
- dectermport failed to find language ME 2-21
- delete_file I/O error, VMS Status ME 2-23
- directory not empty (ENOTEMPTY) ME 2-23
- disk
 - full writing setup file ME 2-23
 - quota exceeded (EDQUOT) ME 2-24
- dispatcher, bad program #200006 ME 2-24
- don't forget to do a 'KDB_UTIL load' ME 2-24
- duplicate
 - DECnet mapping detected ME 2-25
 - UID ME 2-25
- ECO level 1 was already set in image ME 2-25
- error
 - 65 ME 2-25
 - 84 ME 2-26
 - attempting to change password ME 2-26
 - Couldn't create configuration file ME 2-26
 - NULL token ME 2-27
 - reading new password, password unchanged ME 2-27
 - reading old password ME 2-27
 - receiving startup banner from hostname ME 2-27
 - sen can't \$ASSIGN to FFI device ME 2-28
 - service "nfs" not found ME 2-28
 - updating Kerberos database ME 2-28
 - while deleting directory ME 2-29
 - while deleting mailfile ME 2-29
 - writing output file ME 2-29
- Event Flag Mask is mask_value ME 2-86
- ex0, transmit error=number ME 2-29

- Excelan
 - Receive Errors ME 2-30
 - Transmit Errors ME 2-30
- excessive
 - authentication failures ME 2-31
 - login failures ME 2-31
- EXT_SRVTAB
 - bad instance name ME 2-31
 - couldn't get local realm ME 2-31
- extend_file I/O error, VMS Status ME 2-32
- fatal error user ME 2-32
- file
 - already accessed on channel ME 2-32
 - exists (EEXIST) ME 2-32
- fill_in_file I/O error, VMS Status ME 2-33
- folder
 - can not be deleted ME 2-33
 - context corrupt ME 2-33
 - has inferior folders ME 2-33
 - timer too low ME 2-33
- generic error ME 2-34
- get_buffer Unix error n ME 2-34
- I/O error (EIO) ME 2-36
- incorrect old password ME 2-34
- invalid
 - choose 0-65535 ME 2-34
 - domain literal after @ ME 2-35
 - group mailbox list ME 2-35
 - mailbox list ME 2-35
 - setup file format ME 2-35
 - setup filename ME 2-36
- invoked as a type server ME 2-36
- IP Connection is for server ME 2-36
- is a directory (EISDIR) ME 2-37
- KDB_UTIL
 - Couldn't get master key ME 2-37
 - Unable to open filename ME 2-38
- KERBEROS
 - LIST can't find realm of ticket file ME 2-42
- Kerberos
 - Authentication failed ME 2-38
 - db and cache init failed ME 2-39
 - error bad Kerberos name format ME 2-39
 - error error-message ME 2-39
 - error on default value lookup ME 2-40
 - rcmd failed rcmd protocol failure ME 2-42
 - realm ME 2-42
- KERBEROS DATABASE STASH
 - Couldn't read master key ME 2-38
 - Unable to open master key file ME 2-39
 - Write I/O error on master key file ME 2-39
- KERBEROS INIT
 - bad Kerberos instance format ME 2-40
 - bad Kerberos name format ME 2-40
 - bad Kerberos realm format ME 2-40
 - generic error ME 2-41
 - k_gethostname failed ME 2-42
 - krb_get_lrealm failed ME 2-41
 - password incorrect ME 2-41
 - principal unknown (Kerberos) ME 2-41
 - protocol error ME 2-41
 - key file truncated ME 2-42
 - krb_sendauth() failed, principal unknown (Kerberos) ME 2-43
 - LIB\$GET_EF failed for protocol ME 2-43
 - LICENSE-W-NOCOMB ME 2-43
 - line too long before authentication ME 2-43
 - link_file I/O error, VMS Status ME 2-44
 - logical name not defined ME 2-44
 - login
 - failure user ME 2-44
 - user ME 2-45
- MAIL-E-
 - ERRACTRNS ME 2-45
- message from user MULTINET on node
 - dispatcher ME 2-47
- mismatch - try again ME 2-47
- missing
 - address after comma ME 2-47
 - command before authentication host ME 2-47
 - or invalid host name after @ ME 2-47
- mk_req failed, principal unknown ME 2-48
- modify_attributes
 - don't have write access ME 2-49
 - I/O error, VMS Status ME 2-49
- more than 40 found ME 2-50
- mount access denied for client_name ME 2-50
- MultiNet Printer Symbiont
 - Couldn't connect to host ME 2-56
 - error while waiting for ack of CF file ME 2-56
 - Negative acknowledgement ME 2-56, ME 2-57
- MultiNet Server
 - \$CREPRC failed, status = 39c ME 2-58
 - BootP, hardware address not found ME 2-57
 - Couldn't start RPCLOCKMGR ME 2-58
 - DHCP server not starting ME 2-58
 - Failure to create VMS print job = %XC ME 2-59
 - GATED, KERNEL DELETE ME 2-59
 - GATED, krt_delete_dst task ME 2-59
 - GATED, task_send_packet ME 2-59
 - No Program to merge specified for server RPCLOCKMGR ME 2-60
 - R_SERVICES
 - Bogus state dispatch, DCL still running ME 2-60
 - Couldn't create Mailbox ME 2-60
 - gethostbyaddr failed ME 2-60
 - I/O error %MULTINET-W-ECONNCLOSED ME 2-61
 - Socket read error ME 2-61
 - Service

- FINGER pid ME 2-62
- name pid nn failed ME 2-62
- NNTP pid n failed ME 2-62
- Unexpected Exception in
 - MULTINET_SERVER process ME 2-63
- MultiNet SMTP Server Failed to merge user written SMTP customization image ME 2-63
- MULTINET_SHOW
 - can't connect to mount server ME 2-63
 - No MultiNet Kernel ME 2-63
 - Timed out. Resending ME 2-63
- MULTINET-F-
 - ECONNREFUSED ME 2-53
 - ECONNRESET ME 2-53
 - EHOSTUNREACH ME 2-53
- MULTINET-W-STARTUPERR ME 2-56
- must use comma to separate addresses ME 2-64
- named bad referral ('domain' !) ME 2-64
- net
 - read %MULTINET-F-ECONNRESET ME 2-65
 - use failed (code 5) ME 2-65
- new_file out of virtual memory ME 2-65
- new-mail timer too low ME 2-66
- NFS
 - 005F, Authentication failure ME 2-66
 - Server
 - Couldn't allocate miniprocess ME 2-66
 - Couldn't create RPC transport ME 2-66
 - Couldn't create RPC transport stack ME 2-67
 - Error getting socket structure ME 2-67
 - Server/Kernel version mismatch ME 2-67
 - SVC_RECV failed ME 2-67
 - UDP SVC_RECV failed ME 2-67
 - Unexpected Exception in Kernel-Mode ME 2-68
 - Warning writeback cache non-empty ME 2-68
- nfs_read, Unexpected small buffer ME 2-73
- nfs_rename
 - bad context state n ME 2-73
 - channel 0, can't restart ME 2-73
- no
 - buffer space available (ENOBUFFS) ME 2-74
 - space left on device (ENOSPC) ME 2-74
 - subscriptions ME 2-74
 - such file or directory (ENOENT) ME 2-74
 - tickets
 - in file ME 2-75
 - to destroy ME 2-75
- non-authoritative answer ME 2-75
- not
 - a directory (ENOTDIR) ME 2-75
 - an exported filesystem for name ME 2-75
 - owner (EPERM) ME 2-76
- null
 - command before authentication host ME 2-76
 - passwords are not allowed ME 2-76
- open_decnet
 - \$ASSIGN error ME 2-81
 - \$QIOW error ME 2-81
 - \$QIOW IOSB error ME 2-81
- open_tcp
 - connect error ME 2-82
 - setsockopt error ME 2-82
 - socket error ME 2-82
 - Unable to resolve IP address for X server node ME 2-82
- opening
 - DECnet Connection To ME 2-80
 - value connection to Node name, Server ME 2-81
- operation not supported on socket (EOPNOTSUPP) ME 2-82
- out of free storage ME 2-83
- pad, padding_amount ME 2-83
- password NOT changed ME 2-83
- permission denied (EACCES) ME 2-84
- Printer Server
 - %RMS-F-RFA, invalid record's file address (RFA) ME 2-85
 - ansi_q, Recvjob lost connection Error ME 2-84
 - Failed to merge user written LPD Server image ME 2-85
- probable bogus newsgroup list ME 2-85
- protocol Event Flag is flag_value ME 2-85
- read only file system (EROFS) ME 2-88
- RMT-I-REMINFO, Remote error code 34516 ME 2-89
- RPC timed out server not responding ME 2-90
- RPCMount
 - Refused RPCMount request ME 2-91
 - Rejected mount request from client_name
 - Couldn't get file handle for mount_point ME 2-91
 - non-AUTH_UNIX credentials ME 2-91
 - unable to get hostname for ip_address ME 2-92
 - impostor ME 2-92
 - ip_address
 - mount_point is not an exported filesystem ME 2-92
 - mount_point is not an exported file system for client_name ME 2-92
- se0, Transmit error ME 2-93

-
- setsockopt(SO_RCVBUF) ME 2-94
 - setup file name not found ME 2-94
 - SHOW-W-OPENIN ME 2-98
 - signal caught ME 2-94
 - stale file system (ESTALE) ME 2-95
 - start_decnet \$QIOW error ME 2-95
 - start_tcp \$QIOW error ME 2-95
 - startup message, Debug level is value ME 2-95
 - syslog MultiNet Server pausing for old server to write
dump file and exit ME 2-96
 - SYSTEM-F-
 - BADVEC ME 2-98
 - FORCEDEXIT ME 2-98
 - IVDEVNAM ME 2-98
 - VASFULL ME 2-99
 - tcp/ip remote startup error timeout occurred
connection to node nodename ME 2-100
 - telnet, out of space ME 2-100
 - text file busy (ETXTBSY) ME 2-100
 - there were more tuples found than there were space
for ME 2-101
 - tickets
 - destroyed ME 2-101
 - NOT destroyed ME 2-101
 - too many open files (EMFILE) ME 2-101
 - unable to
 - create
 - scratch file to write message data
ME 2-103
 - TCP socket error message ME 2-103
 - get local realm ME 2-104
 - get peer name error message ME 2-104
 - init network channel ME 2-104
 - unchanged ME 2-104
 - unexpected
 - characters
 - after address in group ME 2-104
 - at end of address ME 2-105
 - VMS error, SS\$_EXQUOTA ME 2-105
 - unknown character set ME 2-105
 - unparseable date field ME 2-105
 - unterminated
 - comment ME 2-105
 - mailbox ME 2-106
 - user authorization failure ME 2-106
 - warning
 - message has unknown MIME version
ME 2-106
 - unexpected error 9 (ffff)Good Evening
ME 2-106
 - X Toolkit Error Can't Open display ME 2-107
 - X11DEBUG
 - F-NOSERVER ME 2-107
 - I-USERACTION, either the server is
down ME 2-108
 - Xgateway
 - \$ASSIGN to _NET ME 2-109
 - Cannot Open DECnet Channel to Node name
Server number ME 2-109
 - Configuration Error - value Logical Name Not
Defined ME 2-108
 - IP connection failed to node name server
number ME 2-109
 - IP setsockopt call failed ME 2-109
 - IP socket call failed ME 2-108
 - Unable to resolve IP address for X server node
hostname ME 2-108
 - EXIT-ON-EXCEPTION AD 19-46
 - EXPAN AD 8-10
 - export list AD 19-17
 - expressions AD 12-48
 - Boolean AD 12-48
 - data AD 12-49
 - numeric AD 12-51
 - extended SMTP (RFC-1869) AD 8-2
-
- ## F
- FAQs AD 1-6
 - fax AD 1-5
 - file name
 - defining AD 11-10
 - mapping AD 20-6
 - file systems on Sun hosts AD 19-20
 - FILECACHE-DEBUG AD 19-46
 - files
 - SNMPD.CONF AD 15-4
 - SNMPSERVER.LOG AD 15-9
 - TEMPLATE_SNMPD.CONF AD 15-7
 - firewalls IN 1-18
 - configuring IN 1-18
 - firewalls, transferring files from UG 6-15
 - font
 - catalogues AD 14-9
 - data AD 14-6
 - server AD 14-1
 - adding fonts to the AD 14-10
 - cache AD 14-9
 - configuration checking AD 14-5
 - configuration parameter
 - alternate-server AD 14-2
 - cache size AD 14-2
 - catalogue AD 14-2
 - client-limit AD 14-2
 - default-point-size AD 14-2
 - default-resolutions AD 14-3
 - error-file AD 14-3
 - port AD 14-3
 - trusted-clients AD 14-3
 - types AD 14-4

- format of
 - COUNTRY specification AD 7-3
 - RULE specification AD 7-4
 - ZONE specification AD 7-4
- forward
 - first AD 6-11
 - only AD 6-11
- forwarded ports
 - tunnels UG 8-8
- forwarders AD 6-11
- FTP
 - anonymous UG 6-14, AD 11-2
 - client, configuring the AD 11-1
 - command
 - ACCOUNT UG B-6
 - AGET UG B-7
 - APPEND GET UG B-8
 - APPEND PUT UG B-9
 - APPEND RECEIVE UG B-10
 - APPEND SEND UG B-11
 - APUT UG B-12
 - ASCII UG B-13
 - ATTACH UG B-14
 - BELL UG B-15
 - BINARY UG B-16
 - BLOCK UG B-17
 - BYE UG B-18
 - BYTE UG B-19
 - CD UG B-20
 - CDUP UG B-21
 - CLOSE UG B-22
 - CONFIRM UG B-23
 - CONNECT UG B-24
 - CPATH UG B-25
 - CREATE-DIRECTORY UG B-26
 - CWD UG B-27
 - DELETE UG B-28
 - DIRECTORY UG B-29
 - DISCONNECT UG B-30
 - EXIT UG B-31
 - EXIT-ON-ERROR UG B-32
 - GET UG B-33
 - HASH UG B-34
 - HELP UG B-35
 - LCD UG B-36
 - LDIR UG B-37
 - LIST UG B-38
 - LOCAL-CD UG B-39
 - LOCAL-DIRECTORY UG B-40
 - LOCAL-PWD UG B-41
 - LOGIN UG B-42
 - LPWD UG B-43
 - LS UG B-44
 - MDELETE UG B-45
 - MGET UG B-46
 - MKDIR UG B-47
 - MPUT UG B-48
 - MULTIPLE DELETE UG B-49
 - MULTIPLE GET UG B-50
 - MULTIPLE PUT UG B-51
 - MULTIPLE RECEIVE UG B-52
 - MULTIPLE SEND UG B-53
 - OPEN UG B-54
 - PASSIVE UG B-55
 - PASSWORD UG B-57
 - PORT UG B-58
 - PROMPT-FOR-MISSING-ARGUMENTS UG B-59
 - PROMPT-ON-CONNECT UG B-60
 - PUSH UG B-61
 - PUT UG B-62
 - PWD UG B-64
 - QUIT UG B-65
 - QUOTE UG B-66
 - RECEIVE UG B-67
 - RECORD-SIZE UG B-68
 - REMOTE-HELP UG B-69
 - REMOVE-DIRECTORY UG B-70
 - RENAME UG B-71
 - RETAIN UG B-72
 - RM UG B-73
 - RMDIR UG B-74
 - SEND UG B-75
 - SET UG B-76
 - SHOW-DIRECTORY UG B-77
 - SITE UG B-78
 - SPAWN UG B-79
 - STATISTICS UG B-80
 - STATUS UG B-81
 - STREAM UG B-82
 - STRUCTURE UG B-83
 - TAKE UG B-84
 - TENEX UG B-85
 - TYPE UG B-86
 - USER UG B-87
 - VERBOSE UG B-88
 - VERSION UG B-89
 - command scripts UG 6-13
 - initialization file UG 6-15
 - log files UG 6-14, AD 11-4
 - messages, defining AD 11-9
 - security, managing AD 11-5
 - server
 - login command procedure AD 11-3
 - managing an AD 11-1
 - qualifiers AD 11-4
 - server connection banner AD 11-6
 - site command
 - SITE +VMS+ AD 11-9
 - SITE NONE AD 11-9
 - SITE PRIV AD 11-9
 - SITE RMS RECSIZE AD 11-9

SITE RMS STREAM AD 11-9
 SITE SHOW TIME AD 11-9
 SITE SPAWN AD 11-9
 SITE VMS AD 11-9
 SITE WINDOW-SIZE AD 11-9
 using AD 11-8
 troubleshooting UG 6-16
 using commands UG 6-5
 VMS structure UG 6-11
 fully qualified domain name (FQDN) AD 6-10

G

GATED IN 6-10
 configuration file syntax AD 5-4
 configuring AD 5-3
 enabling AD 5-4
 implementation notes AD 5-14
 primitives AD 5-5
 protocol configuration AD 5-9
 trace options AD 5-7
 GENERIC AD 19-30
 gethostbyaddr() AD 6-9
 gethostbyaddr() PR 2-2
 gethostbyname() AD 6-9
 gethostbyname() PR 2-2, PR 2-3, PR 3-1
 getservbyname() PR 2-2, PR 2-3, PR 2-4
 getservbyport() PR 2-2
 global parameters AD 3-47
 guide contents AD 1-1

H

h_errno PR 3-10
 hardware clock AD 7-1
 HELLO protocol
 configuring AD 5-10
 HELP AR 10-14
 HIBERNATE-ON-EXCEPTION AD 19-46
 home directory AD 21-10
 host
 alias specifying UG 3-3
 definitions, adding AD 6-5
 equivalences UG 5-3
 information, displaying UG 2-2
 name
 conformance AD 6-5
 generation AD 12-26
 table AD 6-2, AD 6-6
 configuring AD 6-3
 source files AD 6-2
 tables information AD 6-3
 host key

 public part AD 21-20
 host name patterns AD 21-7
 HOST.EQUIV UG 5-4
 HOST_TABLE COMPILE AD 6-7
 HOST-ALIAS-FILE AD 8-15
 HOSTS.EQUIV AD 4-20
 htonl() PR 2-2
 hton() PR 2-2

I

IBM 3278 models UG 5-10
 ICMP redirect handling
 configuring AD 5-12
 idle timeout AD 21-8
 ignore AD 7-20
 IMAP
 directives file AD 8-19
 mail folders AD 8-18
 options in the global .IMAPRC file AD 8-20
 server AD 8-18
 state information files AD 8-21
 supported logical AD 8-21
 INADDR_ANY PR 2-4, PR 2-5
 incomplete mappings AD 19-6
 individual aliases, specifying UG 3-3
 inet_addr() PR 2-2
 inet_ntoa() PR 2-2
 Input/Output Status Block (IOSB) PR 4-1
 insecure network UG 8-1
 installation dialog IN 2-4
 interfaces and parameters AD 3-10
 dn AR 5-4
 nsip AR 5-4
 pd AR 5-5
 ppp AR 5-5
 psi AR 5-6
 rp AR 5-6
 se AR 5-7
 sl AR 5-7
 Internet Time Servers (ITSs) AD 7-13
 interrupt vectors AD 3-13
 intruders AD 4-12
 IO\$M_EXTEND PR 2-6
 IP
 address pool availability AD 12-79
 client access to a DECnet server AD 17-2
 connectivity AD 3-2
 transport, configuring IN 1-17
 error message
 named lame server on 'domain' (in 'domain?')
 ME 2-64
 IP-CLUSTER-ALIASES AD 3-49
 IP-over-DECnet circuits AD 3-35

IP-over-PSI configuration AD 3-36
ISC BIND 8.2.3 Nameserver AD 6-9

K

KDCs AD 16-15
keepalive UG 8-13
keepalive messages AD 21-8
keepalive timers AD 4-17
Kerberos AD 16-1
 administration AD 16-11
 database AD 16-3
 EDIT prompts AD 16-8
 password, changing UG 4-3
 tickets AD 16-2
 understanding UG 4-1
keyboard mapping file format UG 5-13
keyword value pairs AD 21-4

L

lease information for
 all leased IP addresses AD 12-76
 specific IP addresses AD 12-76
license PAK
 installing IN 2-1
 registering and loading IN 2-1
limited AD 7-21
listen() PR 2-4
loadable timezone rules AD 7-3, AD 7-5
local mail AD 8-14
local-master AD 7-15
log file name, specifying AD 11-10
logical
 DECW\$DISPLAY UG 8-4
 MULTINET AD 2-4
 MULTINET_ACCESS_CHALLENGE_FORMAT
 AD 4-30, AD 4-31
 MULTINET_ACCESS_OTP_FORMAT AD 4-31
 MULTINET_ACCESS_RESPONSE_FORMAT
 AD 4-31
 MULTINET_ANONYMOUS_FTP_CONTROL
 AD 11-3
 MULTINET_ANONYMOUS_FTP_DIRECTORY
 AR 5-29
 MULTINET_ANONYMOUS_PASSWORD AD 4-13,
 AD 11-4
 MULTINET_CLUSTER_SERVICE_ADDRESS
 AR 5-30
 MULTINET_CLUSTER_SERVICE_NAMES AR 5-31
 MULTINET_COMMON_ROOT IN 1-8, IN 1-13
 MULTINET_DIRECTORY_MESSAGE_FILENAME
 AD 11-10

MULTINET_DISABLE_SPAWN UG B-14, UG B-61,
 UG B-79, UG C-5, UG C-19, UG C-35, DN A-8,
 DN A-16, DN A-23, AR 3-4, AR 3-13, AR 3-17,
 AR 4-8, AR 4-17, AR 4-43, AR 5-12, AR 5-26,
 AR 5-53, AR 6-13, AR 6-26, AR 6-55, AR 8-5,
 AR 8-10, AR 8-20, AR 9-6, AR 9-15, AR 9-47,
 AR 10-6, AR 10-16, AR 10-59
MULTINET_FTP_221_REPLY AD 11-9
MULTINET_FTP_421_REPLY AD 11-9
MULTINET_FTP_ADDRESS AD 4-13, AD 11-4
MULTINET_FTP_ALL_VERSIONS AD 11-7
MULTINET_FTP_CONNECT_BANNER AD 11-6
MULTINET_FTP_DODROP1DOT AD 11-8
MULTINET_FTP_FAST_TIMEOUT AD 11-8
MULTINET_FTP_HOSTNAME AD 4-13, AD 11-4
MULTINET_FTP_INCLUDE_DEVICE_IN_NLST
 AD 11-8
MULTINET_FTP_LOCAL_ADDRESS AD 11-4
MULTINET_FTP_LOG_ALL_USERS AD 11-10
MULTINET_FTP_LOGFILE AD 11-10
MULTINET_FTP_MAXIMUM_IDLE_TIME AD 4-13,
 AD 11-8
MULTINET_FTP_NONPASV UG 6-15
MULTINET_FTP_PASSWORD_WARNING_
 MESSAGE AD 11-10
MULTINET_FTP_PASSWORD_WARNING_TIME
 AD 11-10
MULTINET_FTP_PWDEXPIRED AD 11-11
MULTINET_FTP_PWDPREEXP AD 11-11
MULTINET_FTP_SERVER_LOG_LIMIT AD 11-5
MULTINET_FTP_STRIP_VERSION AD 11-6,
 AD 11-7
MULTINET_FTP_SYST_BANNER AD 11-7
MULTINET_FTP_UNIX_STRIP_VERSION AD 11-6
MULTINET_FTP_UNIX_STYLE_BY_DEFAULT
 AD 11-6
MULTINET_FTP_UNIX_STYLE_CASE_
 INSENSITIVE AD 11-7
MULTINET_FTP_UNIX_YEAR_OLD_FILES
 AD 11-8
MULTINET_FTP_WINDOW_SIZE UG 6-6
MULTINET_HOST_ALIAS_FILE UG 3-3
MULTINET_HOST_NAME PR 3-20, AR 5-37
MULTINET_HOSTALIASES PR 3-34
MULTINET_IP_CLUSTER_ALIASES AD 3-49,
 AR 5-38
MULTINET_KRBTKT_username AD 16-3
MULTINET_LOCALDOMAIN AR 5-42
MULTINET_LPD_DEFAULT_USERNAME AR 5-43
MULTINET_NAMESERVER_RETRANS AD 11-8,
 AR 5-44
MULTINET_NAMESERVER_RETRY AD 11-8,
 AR 5-44
MULTINET_NAMESERVERS AD 6-9, AD 6-10,
 AD 6-12, AR 5-35
MULTINET_NETWORK_IMAGE PR 3-47

-
- MULTINET_NFS_SERVER_NFS_ACL_SUPPORT_DISABLED AD 19-28
 - MULTINET_NTYSMB AD 9-20
 - MULTINET_PCNFSD_PRINTER_LIMIT AD 19-30
 - MULTINET_PCNFSD_QUEUE_TYPES AD 19-30
 - MULTINET_POPx_DEST_FOLDER AD 8-23
 - MULTINET_POPx_SOURCE_FOLDER AD 8-23
 - MULTINET_RCP_INDEX_UPTO_EOF UG 6-1
 - MULTINET_RMT_TAPE_DEVICE AR 5-33
 - MULTINET_ROOT IN 1-8, IN 1-13, AD 2-4
 - MULTINET_SEARCHDOMAINS AD 6-12
 - MULTINET_SERVER ME 1-2
 - MULTINET_SERVER_NOACNT AD 2-2
 - MULTINET_SMTP_A1_DOMAIN AD 8-26
 - MULTINET_SMTP_A1_NAME AD 8-26
 - MULTINET_SMTP_ACCEPT_UNIX_LF AD 8-9
 - MULTINET_SMTP_ACCEPT_UNIX_LF_BRAIN_DAMAGE AD 8-9
 - MULTINET_SMTP_AM_DOMAIN AD 8-26
 - MULTINET_SMTP_AM_NAME AD 8-26
 - MULTINET_SMTP_APPEND_FORWARDER_TO_MX AD 8-14
 - MULTINET_SMTP_BATCH_QUEUE AD 8-2
 - MULTINET_SMTP_DISABLE_FOLDER_DELIVERY AD 8-2
 - MULTINET_SMTP_FROM_HOST UG 3-3, AD 8-15
 - MULTINET_SMTP_HOST_NAME UG 3-3
 - MULTINET_SMTP_MAXIMUM_822_TO_LENGTH AD 8-9
 - MULTINET_SMTP_MRGATE-NAME AD 8-26
 - MULTINET_SMTP_REJECT_INVALID_DOMAINS AD 8-9
 - MULTINET_SMTP_REPLY_TO AD 8-7, AD 8-10, AR 4-3, AR 4-25
 - MULTINET_SMTP_SERVER_DISABLE_VRFYEXPN AD 8-10
 - MULTINET_SMTP_SERVER_REJECT_FILE AD 8-3
 - MULTINET_SMTP_SERVER_REJECT_INFO AD 8-6
 - MULTINET_SMTP_SUPPRESS_VENDOR AD 8-9
 - MULTINET_SOCKET_LIBRARY AD 6-8
 - MULTINET_SPOOL AR 5-45
 - MULTINET_SSH_ALLOW_EXPIRED_PW AD 21-22
 - MULTINET_SSH_ALLOW_PREEXPRIED_PW AD 21-23
 - MULTINET_SSH_KEYGEN_MIN_PW_LEN AD 21-23
 - MULTINET_SSH_PARAMETERS AD 21-23
 - MULTINET_SSH_USE_SYSGEN_LGI AD 21-23
 - MULTINET_STREAM_SYMBIONT_TIMERS AD 9-11
 - MULTINET_TELNET_PRINT_ESCAPE_CHARACTER UG A-32
 - MULTINET_TFTP_DEFAULT_DIRECTORY AR 5-46
 - MULTINET_TN3270_APPLICATION_KEYPAD UG 5-19
 - MULTINET_TN3270_LANGUAGE UG 5-19, UG 5-20
 - MULTINET_TN3270_PRINTER UG 5-18
 - MULTINET_TN3270_TRANSLATION_TABLES UG 5-20
 - MULTINET_TN5250_APPLICATION_KEYPAD UG 5-19
 - MULTINET_TN5250_PRINTER UG 5-18
 - MULTINET_VMSMAIL_LOCASE_USERNAME AD 8-9
 - MULTINET_VMSMAIL_USE_RFC822_TO_HEADER AD 8-17
 - MULTINET_WHOIS_DEFAULT_SERVER AR 5-49
 - MULTINET_XGATEWAY_DEBUG_LEVEL AD 17-7
 - MULTINET_XGATEWAY_DECNET_server_number_HOSTNAME AD 17-5
 - MULTINET_XGATEWAY_DECNET_server_number_SERVER AD 17-5
 - MULTINET_XGATEWAY_TCPIP_server_number_HOSTNAME AD 17-3
 - MULTINET_XGATEWAY_TCPIP_server_number_SERVER AD 17-3
 - SSH_DIR AD 21-21
 - SSH_EXE AD 21-22
 - SSH_LOG AD 21-22
 - SSH_MAX_SESSIONS AD 21-22
 - SSH_TERM_MBX AD 21-22
 - logicals
 - defining them system-wide AD 8-24
 - login timeout, changing the AD 4-28
 - LOGIN.COM, inhibiting output from UG 6-3
 - loopstats AD 7-18
 - LPD
 - and stream symbiont AD 9-15
 - jobs (inbound) AD 9-17
 - print jobs AD 9-5
 - protocol queue AD 9-8
 - server AD 9-1, AD 9-5
 - spool directory AD 9-4
 - lpr -v support AD 9-16
 - LPR/LPD server AD 9-1
-
- ## M
- mail
 - alias file AD 8-17
 - aliases AD 8-16
 - delivery mechanisms AD 8-2
 - gateways AD 8-14
 - hub AD 8-11
 - messages AD 8-3
 - parameter
 - ALIAS-FILE AD 8-7
 - DECNET-DOMAIN AD 8-7
 - DELIVERY-RECEIPTS AD 8-7

- DISABLE-PSMAIL AD 8-7
- DISALLOW-USER-REPLY-TO AD 8-7
- FORWARDER AD 8-7
- FORWARD-LOCAL-MAIL AD 8-7
- FORWARD-REMOTE-MAIL AD 8-7
- HEADER-CONTROL AD 8-7
- HOST-ALIAS-FILE AD 8-7
- LOCAL-MAIL-FORWARDER AD 8-7
- POSTMASTER AD 8-8
- QUEUE-COUNT AD 8-8
- REPLY-CONTROL AD 8-8
- RESENT-HEADERS AD 8-8
- RETRY-INTERVAL AD 8-8
- RETURN-INTERVAL AD 8-8
- SEND-BROADCAST-CLASS AD 8-8
- SMTP-HOST-NAMES AD 8-8
- START-QUEUE-MANAGER AD 8-8
- parameters with MAIL-CONFIG AD 8-6
- queues AD 8-9, AD 8-10
- mailbox AD 21-22
- MAILbus AD 8-31
- MAIL-CONFIG command
 - ADD GATEWAY AR 4-5
 - ADD LOCAL-DOMAIN AR 4-6
 - ADD QUEUE-GROUP AR 4-7
 - ATTACH AR 4-8
 - CLEAR AR 4-9
 - DELETE GATEWAY AR 4-10
 - DELETE LOCAL-DOMAIN AR 4-11
 - DELETE QUEUE-GROUP AR 4-12
 - ERASE AR 4-13
 - EXIT AR 4-14
 - GET AR 4-15
 - HELP AR 4-16
 - PUSH AR 4-17
 - QUIT AR 4-18
 - REMOVE GATEWAY AR 4-19
 - REMOVE QUEUE-GROUP AR 4-20
 - SAVE AR 4-21
 - SET ALIAS-FILE AR 4-22
 - SET DECNET-DOMAIN AR 4-23
 - SET DELIVERY-RECEIPTS AR 4-24
 - SET DISABLE-PSMAIL AR 4-26
 - SET DISALLOW-USER-REPLY-TO AR 4-25
 - SET FORWARDER AR 4-27
 - SET FORWARD-LOCAL-MAIL AR 4-28
 - SET FORWARD-REMOTE-MAIL AR 4-29
 - SET HEADER-CONTROL AR 4-30
 - SET HOST-ALIAS-FILE AR 4-31
 - SET LOCAL-MAIL-FORWARDER AR 4-32
 - SET POSTMASTER AR 4-33
 - SET QUEUE-COUNT AR 4-34
 - SET REPLY-CONTROL AR 4-35
 - SET RESENT-HEADERS AR 4-36
 - SET RETRY-INTERVAL AR 4-37
 - SET RETURN-INTERVAL AR 4-38
 - SET SEND-BROADCAST-CLASS AR 4-39
 - SET SMTP-HOST-NAMES AR 4-40
 - SET START-QUEUE-MANAGER AR 4-41
 - SHOW AR 4-42
 - SPAWN AR 4-43
 - STATUS AR 4-44
 - USE AR 4-45
 - VERSION AR 4-46
 - WRITE AR 4-47
- mailing lists AD 8-16
- martian networks AD 5-9
- master name server AD 6-12
- master-clock AD 7-15
- matched leases for
 - client ID AD 12-78
 - hardware addresses AD 12-78
- maximum idle time, specifying AD 11-8
- MENU-CONFIG AD 2-10, AD 4-7, AD 4-30
 - command AD 6-12
 - using DN 2-2
- methods of associating IP addresses and host names AD 6-1
- monitor AD 7-16
- MOP AD 19-32
- MOUNT AR 1-80
- mount
 - parameter settings AD 19-21
 - point
 - option usage AD 19-35
 - options AD 19-35
 - points, naming AD 19-16
- MTU discovery AD 3-50
- multicast support AD 3-50
- MULTINET
 - HOSTS.EQUIV UG 5-3, UG 8-1, AD 4-20, AD 10-1
 - SHOST.EQUIV UG 8-1
 - SHOW/SNMP
 - commands AD 15-10
 - /ARP AD 15-10
 - /CONNECTIONS AD 15-10
 - /MIB_VAR AD 15-10
 - /ROUTE AD 15-10
 - /STATISTICS AD 15-10
 - SSH_KNOWN_HOSTS AD 21-18
 - SSHD_CONFIG AD 21-4
- MultiNet
 - command-line interface configuration tools AD 2-4
 - configuration file summary AD 2-5
 - console messages ME 1-1
 - definition IN 6-1
 - de-installation command procedure IN 4-1
 - directory
 - layout IN 1-8
 - structure IN 1-9
 - disk space requirements IN 1-6
 - distribution media IN 1-5

- documentation
 - set IN 5-1
- error code
 - E2BIG ME B-1
 - EACCES ME B-1
 - EADDRINUSE ME B-1
 - EADDRNOTAVAIL ME B-1
 - EAFNOSUPPORT ME B-1
 - EAGAIN ME B-1
 - EALREADY ME B-1
 - EBADF ME B-1
 - EBUSY ME B-1
 - ECHILD ME B-1
 - ECONNABORTED ME B-1
 - ECONNREFUSED ME B-2
 - ECONNRESET ME B-2
 - EDEADLK ME B-2
 - EDESTADDRREQ ME B-2
 - EDOM ME B-2
 - EDQUOT ME B-2
 - EEXIST ME B-2
 - EFAULT ME B-2
 - EFBIG ME B-2
 - EHOSTDOWN ME B-2
 - EHOSTUNREACH ME B-2
 - EINPROGRESS ME B-2
 - EINTR ME B-2
 - EINVAL ME B-2
 - EIO ME B-3
 - EISCONN ME B-3
 - EISDIR ME B-3
 - ELOOP ME B-3
 - EMFILE ME B-3
 - EMLINK ME B-3
 - EMSGSIZE ME B-3
 - ENAMETOOLONG ME B-3
 - ENETDOWN ME B-3
 - ENETRESET ME B-3
 - ENETUNREACH ME B-3
 - ENFILE ME B-3
 - ENOBUFS ME B-3
 - ENODEV ME B-4
 - ENOENT ME B-4
 - ENOEXEC ME B-4
 - ENOMEM ME B-4
 - ENOPROTOOPT ME B-4
 - ENOSPC ME B-4
 - ENOTBLK ME B-4
 - ENOTCONN ME B-4
 - ENOTDIR ME B-4
 - ENOTEMPTY ME B-4
 - ENOTSOCK ME B-4
 - ENOTTY ME B-4
 - ENXIO ME B-4
 - EOPNOTSUPP ME B-4
 - EPERM ME B-4
 - EPFNOSUPPORT ME B-4
 - EPIPE ME B-4
 - EPROCLIM ME B-4
 - EPROTONOSUPPORT ME B-4
 - EPROTOTYPE ME B-4
 - ERANGE ME B-4
 - EROFS ME B-4
 - ESHUTDOWN ME B-4
 - ESOCKTNOSUPPOT ME B-4
 - ESPIPE ME B-4
 - ESRCH ME B-5
 - ETIMEDOUT ME B-5
 - ETOOMANYREFS ME B-5
 - ETXTBSY ME B-5
 - EUSERS ME B-5
 - EVMISERR ME B-5
 - EWOULDBLOCK ME B-5
 - EXDEV ME B-5
- error messages ME 1-3, ME 2-1
- font server AD 14-7
- host tables
 - hosts AD 6-3
 - IP protocol types AD 6-3
 - networks AD 6-3
 - services AD 6-3
- installation
 - steps IN 1-1
- internals, understanding IN 7-2
- IP transport parameter checklist IN 1-3
- log files ME 1-2
- NFS client, use of user IDs AD 20-2
- online help IN 5-10
- organization IN 7-3
- problem solving ME 1-1
- programming tokens AD 4-31
- public mailing list AD 1-6
- release notes IN 1-5
- Secure Shell (SSH) client UG 8-1
- Secure/IP AD 3-27, AD 3-32, AD 4-24
 - commands for programming tokens AD 4-34
 - commands for testing tokens AD 4-34
 - configuration checklist AD 4-25
 - software requirements AD 3-33
 - terminology AD 3-34
 - using AD 4-34
- Secure/IP, installing IN 1-17
- supported
 - devices IN 7-1
 - protocols IN 7-2
- system
 - disk back up IN 1-5
- updating system parameters IN 1-7
- user
 - profile database, viewing AD 4-31
 - profiles, adding and modifying AD 4-32
 - profiles, managing AD 4-31

XDM server AD 13-3
MULTINET BOOTP-SERVER.CONFIGURATION
 AD 12-16
MULTINET CHECK AR 1-6
MULTINET CONFIGURE AR 1-7
MULTINET CONVERT_UNIX_HOST_TABLE.COM
 AD 6-6
MULTINET DHCPD.CONF AD 12-16
MULTINET DHCPD.LEASES AD 12-16
MULTINET DHCP-SERVER.CONFIGURATION
 AD 12-16
MULTINET DHCP-STATE.DAT AD 12-16
MULTINET DIG AR 1-10
MULTINET DNSKEYGEN AR 1-16
MULTINET DNSSIGNER AR 1-19
MULTINET FONT_COMPILE AR 1-26
MULTINET FONT_INFO AR 1-27
MULTINET FONT_LIST AR 1-28
MULTINET FONT_MKFONTDIR AR 1-29
MULTINET FONT_SHOW AR 1-30
MULTINET FONT_UNCOMPILE AR 1-32
MULTINET HOST_TABLE_COMPILE AR 1-33
MULTINET HOST_TABLE_GET AR 1-35
MULTINET HOST_TABLE_INSTALL AR 1-37
MULTINET HOSTS.LOCAL AD 6-3, AD 6-10
MULTINET HOSTS.SERVICES AD 6-3
MULTINET INSTALL_DATABASES AD 6-7
MULTINET KERBEROS DATABASE DUMP AR 1-38
MULTINET KERBEROS DATABASE EDIT AR 1-39
 prompts AR 1-39
MULTINET KERBEROS DATABASE INITIALIZE
 AR 1-42
MULTINET KERBEROS DATABASE LOAD AR 1-43
MULTINET KERBEROS DATABASE
 NEW_MASTER_KEY AR 1-44
MULTINET KERBEROS DATABASE SRVTAB AR 1-45
MULTINET KERBEROS DATABASE STASH AR 1-46
MULTINET LOAD AR 1-47
MULTINET NAMED.CONF AD 6-13
 file AD 6-9, AD 6-13
MULTINET NETCONTROL AR 1-48
MULTINET NETWORK_DATABASE AD 6-2, AD 6-7
MULTINET NFSDISMOUNT AR 1-61
MULTINET NFSMOUNT AR 1-62
MULTINET NSLOOKUP AR 1-68
MULTINET PING AR 1-75
 OpenVMS status code
 SS\$_DATA_LOST ME A-1, AR 1-75
 SS\$_IVBUFLN ME A-1, AR 1-75
 SS\$_NOPRIV ME A-1, AR 1-75
 SS\$_NORMAL ME A-1, AR 1-75
 SS\$_NOSUCHNODE ME A-1, AR 1-75
 SS\$_PROTOCOL ME A-1, AR 1-75
 SS\$_UNREACHABLE ME A-1, AR 1-75
MULTINET PROFILE /DELETE AR 2-3
MULTINET PROFILE /MODIFY AR 2-4
MULTINET PROFILE /SHOW AR 2-5
MULTINET PROFILE /SUMMARY AR 2-6
MULTINET RCP completion code
 RCP\$COPIED ME A-2
 RCP\$CREATED ME A-2
 RCP\$CREATEDIR ME A-2
 RCP\$ERROR ME A-2
 RCP\$FATALERR ME A-2
 RCP\$LOSTCONN ME A-2
 RCP\$NEWFILES ME A-2
 RCP\$NOSUCHNODE ME A-2
 RCP\$NOTCONNECTED ME A-2
 RCP\$NOTIMPL ME A-2
 RCP\$OPENIN ME A-2
 RCP\$OPENOUT ME A-2
 RCP\$PARSERR ME A-2
 RCP\$PROTOCOLERR ME A-2
 RCP\$SIZECHANGE ME A-2
 RCP\$STARTUPERR ME A-2
 RCP\$WRITERR ME A-2
MULTINET RDATE AR 1-77
MULTINET RMTALLOC AR 1-78
MULTINET RMTALLOC BLOCKSIZE AR 1-80
MULTINET RMTALLOC BROKEN AR 1-81
MULTINET RMTALLOC COMMENT AR 1-80
MULTINET RMTALLOC DENSITY AR 1-80
MULTINET RMTALLOC LABEL AR 1-80
MULTINET RMTALLOC UNIX AR 1-81
MULTINET RWALL AR 1-85
MULTINET SET /ARP AR 1-86
MULTINET SET /DECNET AD 18-4, AR 1-88
MULTINET SET /INTERFACE AR 1-90
MULTINET SET /ROUTE AR 1-95
MULTINET SET /TIMEZONE AR 1-97
MULTINET SHOW AR 1-98
MULTINET SKEY AR 2-7
MultiNet SSH server AD 21-1
MULTINET SSHADD UG 8-21
MULTINET SSHAGENT UG 8-20
MULTINET SSHKEYGEN UG 8-22
MULTINET START_SERVER AD 6-7
MULTINET START_SMTP AD 6-8
MULTINET TCPDUMP AR 1-105
MULTINET TCPVIEW AR 1-109
MULTINET TCPVIEW MENU
 CAPTURE AR 1-111
 FILE AR 1-111
 FILTER AR 1-111
 HELP AR 1-113
 OPTIONS AR 1-112
MULTINET TOKEN CRYPTOCARD /CLEAR AR 2-9
MULTINET TOKEN CRYPTOCARD /LOAD AR 2-10
MULTINET TOKEN CRYPTOCARD /TEST AR 2-19
MULTINET TOKEN SKEY /CLEAR AR 2-20
MULTINET TOKEN SKEY /INITIALIZE AR 2-21
MULTINET TOKEN SKEY /SHOW AR 2-24

MULTINET TOKEN SKEY /TEST AR 2-25
 MULTINET TOKEN SNK /CLEAR AR 2-26
 MULTINET TOKEN SNK /LOAD AR 2-27
 MULTINET TOKEN SNK /TEST AR 2-30
 MULTINET TRACEROUTE AR 1-114
 MULTINET X11DEBUG AR 1-117
 MULTINET_FTP_ANNOUNCE AD 11-6
 MULTINET_NLPx_REMOTE_PRINTER AD 9-10
 MULTINET_SERVER AD 6-7
 multiple
 addresses AD 3-20
 mappings, adding AD 19-15
 print queues, starting AD 9-12
 queues, configuring AD 8-11

N

name mapping DN 2-5
 NAMED.CONF
 options AD 6-18
 zone field
 file AD 6-15
 masters AD 6-15
 type AD 6-15
 name-mapping database, creating DN 2-7
 NCP utility, using DN 1-3
 NET-CONFIG
 command AD 6-12
 interfaces and parameters AR 5-4
 prompts AR 5-7
 ACCM mask AR 5-7
 address and control field compression AR 5-8
 authentication method AR 5-8
 baud rate AR 5-8
 BSD trailer encapsulation AR 5-8
 hardware device AR 5-8
 header compression mode AR 5-8
 ICMP AR 5-9
 idle timeout AR 5-9
 IP address AR 5-9
 IP address of remote system AR 5-9
 IP broadcast address AR 5-9
 IP over DECnet peer host's DECnet name AR 5-9
 IP over PSI local DTE address AR 5-9
 IP over PSI peer DTE address AR 5-9
 IP subnet mask AR 5-9
 link level encapsulation mode AR 5-10
 maximum receive unit (MRU) size AR 5-10
 NetWare link level encapsulation AR 5-10
 NetWare network number AR 5-10
 point-to-point device IP destination address AR 5-10
 protocol compression AR 5-10
 retry count AR 5-10
 termination retry count AR 5-10
 timeout AR 5-11
 VMS device AR 5-11
 NET-CONFIG command
 ADD AR 5-4
 ATTACH AR 5-12
 CHECK AR 5-14
 CLEAR AR 5-17
 DELETE AR 5-18
 DISABLE AR 5-19
 ENABLE AR 5-20
 ERASE AR 5-21
 EXIT AR 5-22
 GET AR 5-23
 HELP AR 5-24
 MODIFY AR 5-25
 PUSH AR 5-26
 QUIT AR 5-27
 SAVE AR 5-28
 SET ANONYMOUS-FTP-DIRECTORY AR 5-29
 SET CLUSTER-SERVICE-ADDRESS AD 6-35, AR 5-30
 SET CLUSTER-SERVICE-NAMES AR 5-31
 SET DEFAULT-RMT-TAPE-DEVICE AR 5-33
 SET DEFAULT-ROUTE AR 5-34
 SET DOMAIN-NAMESERVERS AD 6-8, AD 6-12, AR 5-35
 SET HOST-NAME AR 5-37
 SET IP-CLUSTER-ALIASES AR 5-38
 SET LOAD-EXOS-DRIVER AR 5-39
 SET LOAD-PWIP-DRIVER AR 5-41
 SET LOAD-UCX-DRIVER AR 5-40
 SET LOCAL-DOMAIN AR 5-42
 SET LPD-DEFAULT-USERNAME AR 5-43
 SET NAMESERVER-RETRANSMISSION AD 6-13, AR 5-44
 SET SPOOL-DIRECTORY AR 5-45
 SET TFTP-DIRECTORY AR 5-46
 SET TIMEZONE AR 5-47
 SET TIMEZONE-RULES AR 5-48
 SET WHOIS-DEFAULT-SERVER AR 5-49
 SET WINS-COMPATIBILITY AR 5-50
 SHOW AR 5-51
 SPAWN AR 5-53
 STATUS AR 5-55
 USE AR 5-56
 VERSION AR 5-57
 WRITE AR 5-58
 NETCONTROL
 "R" Server command
 FLUSH-CACHE AR 1-57
 SHOW AR 1-57
 command
 LIST AR 1-50
 NOOP AR 1-50

- QUIT AR 1-50
- QUOTE AR 1-50
- SELECT AR 1-50
- SERVER-VERSION AR 1-50
- STATISTICS AR 1-50
- TIMERS AR 1-51
- VERBOSE AR 1-51
- VERSION AR 1-51
- DHCP command
 - DEBUG AR 1-51
 - DHCP-CONTROL-VERSION AR 1-51
 - DUMP AR 1-51
 - NEWLOG AR 1-51
 - PARTNERDOWN AR 1-51
 - RELEASE AR 1-51
 - RELOAD AR 1-52
 - RESTART AR 1-52
 - SHOW ALL AR 1-52
 - SHOW CID AR 1-52
 - SHOW CLIENT AR 1-52
 - SHOW HADDR AR 1-52
 - SHOW LEASES AR 1-52
 - SHOW POOLS AR 1-52
 - SHOW SUBNET AR 1-52
 - SHUTDOWN AR 1-52
 - START AR 1-52
 - STATISTICS AR 1-53
- DOMAINNAME command
 - DEBUG AR 1-53
 - DUMP AR 1-53
 - MAXIMUM-TTL AR 1-53
 - MINIMUM-TTL AR 1-53
 - QUERYLOG AR 1-53
 - RELOAD AR 1-53
 - RESTART AR 1-53
 - REWRITE-TTL AR 1-53
 - SHOW AR 1-54
 - SHUTDOWN AR 1-54
 - START AR 1-54
 - STATISTICS AR 1-54
 - STOP AR 1-54
 - VERSION AR 1-54
- GATED command
 - DEBUG AR 1-54
 - DUMP AR 1-54
 - TRACE AR 1-55
 - TRACE-FILE AR 1-55
- NFS command
 - ADD MOUNT-RESTRICTION AR 1-55
 - AVERAGE-RESPONSE-TIMES AR 1-55
 - DUMP AR 1-55
 - FILECACHE-DEBUG AR 1-55
 - NFS-CONTROL-VERSION AR 1-55
 - NFSDEBUG AR 1-56
 - RECORD-RESPONSE-TIMES AR 1-56
 - RELOAD AR 1-56
 - RESTART AR 1-56
 - RPCDEBUG AR 1-56
 - SHOW-RESPONSE-TIMES AR 1-56
 - SHUTDOWN AR 1-56
 - START AR 1-56
 - STATISTICS AR 1-56
 - TIMERS AR 1-56
- RARP command
 - DEBUG AR 1-56
 - RELOAD AR 1-56
- RPCMOUNT command
 - CLEAR AR 1-57
 - DEBUG AR 1-57
 - DUMP AR 1-57
 - RELOAD AR 1-57
 - SHOW AR 1-57
- RPCPORTMAP command
 - DEBUG AR 1-57
 - SHOW AR 1-57
- RPCSTATUS command
 - DEBUG AR 1-57
 - RELOAD AR 1-58
 - SHOW AR 1-58
 - SIMULATE-CRASH AR 1-58
- SHOW command AD 12-76
- SSH command
 - DEBUG AR 1-58
 - MASTER_RESTART AR 1-58
 - RESTART AR 1-58
 - SHOW AR 1-59
 - SHUTDOWN AR 1-59
 - START AR 1-59
- STATISTICS command AD 12-80
- TFTP command
 - DEBUG AR 1-59
 - RELOAD AR 1-59
 - SHOW AR 1-59
 - SHOW-TRANSLATION AR 1-59
- UCXQIO command
 - DEBUG AR 1-60
- VIADENET command
 - DEBUG AR 1-60
 - RELOAD AR 1-60
 - SHUTDOWN AR 1-60
- VIAPSI command
 - DEBUG AR 1-60
 - DISCONNECT AR 1-60
 - IDLE AR 1-60
 - RELOAD AR 1-60
 - SHUTDOWN AR 1-60
- netgroups AD 21-21
- network
 - configuration server, choosing AD 12-1
 - definitions AD 6-4
 - interface
 - configuration overview AD 3-2
 - parameters

- ACCM Mask AD 3-6
- Adapter AD 3-6
- Address and Control Field Compression (ACFC) AD 3-7
- Baud Rate AD 3-7
- BSD Trailer Encapsulation AD 3-7
- Communications Mode AD 3-7
- CSR AD 3-7
- Flags AD 3-7
- Hardware Device AD 3-7
- Header Compression Mode AD 3-8
- ICMP AD 3-8
- IP Address AD 3-8
- IP Address of Remote System AD 3-8
- IP Broadcast Address AD 3-8
- IP Over DECnet Peer Host's DECnet Name AD 3-8
- IP Over PSI Local DTE Address AD 3-8
- IP Over PSI Peer DTE Address AD 3-8
- IP SubNet Mask AD 3-9
- Link Level Encapsulation Mode AD 3-9
- Maximum Receive Unit (MRU) Size AD 3-9
- Point-To-Point Device IP Destination Address AD 3-9
- Protocol Compression AD 3-9
- Retry Count AD 3-9
- Termination Retry Count AD 3-9
- Timeout AD 3-9
- Vector AD 3-10
- VMS Device AD 3-9
- interfaces
 - adding AD 3-5
 - with MENU-CONFIG AD 3-15, AD 3-17
 - with NET-CONFIG AD 3-13, AD 3-16
- management station (NMS) AD 15-1
- network interface device drivers IN 7-4
- Network Time Protocol (NTP) AD 7-9
- NFS
 - client AD 20-1, AD 20-13
 - architecture AD 20-8
 - default file attributes AD 20-5
 - mount options AD 20-15
 - semantics AD 20-16
 - setup AD 20-8
 - systems for UID/GID mappings AD 19-5
 - using BACKUP AD 20-15
 - file access, controlling AD 19-22
 - groups AD 20-11
 - adding and deleting AD 19-14
 - mode of operation AD 19-36
 - server AD 19-2
 - architecture AD 19-9
 - configuration AD 19-10
 - global parameters AD 19-36
 - memory AD 19-37
 - mount point options AD 19-34
 - server's ACL support AD 19-28
 - troubleshooting AD 19-46
- NFS-CONFIG AD 19-12
 - utility AD 20-9
- NFS-CONFIG command
 - ADD DECSTATION-MOUNT-POINT AR 6-6
 - ADD EXPORT AR 6-7
 - ADD MOUNT-RESTRICTION AR 6-8
 - ADD NFS-GROUP AR 6-9
 - ADD NFS-PASSWD-FILE AR 6-10
 - ADD UID-TRANSLATION AR 6-11
 - APPEND AR 6-12
 - ATTACH AR 6-13
 - CURRENT AR 6-15
 - DELETE DECSTATION-MOUNT-POINT AR 6-16
 - DELETE EXPORTED-FILE-SYSTEM AR 6-17
 - DELETE MOUNT-RESTRICTION AR 6-18
 - DELETE NFS-GROUP AR 6-19
 - DELETE NFS-PASSWD-FILE AR 6-20
 - DELETE UID-TRANSLATION AR 6-21
 - EXIT AR 6-22
 - GET AR 6-23
 - HELP AR 6-24
 - NETCONTROL AR 6-25
 - PUSH AR 6-26
 - QUIT AR 6-27
 - RELOAD AR 6-28
 - RESTART AR 6-29
 - SAVE AR 6-30
 - SELECT AR 6-31
 - SET APPROXIMATE-TEXT-SIZE-THRESHOLD AR 6-32
 - SET DIRECTORY-INFO-FLUSH-AGE AR 6-33
 - SET DIRECTORY-INFO-IDLE-FLUSH-AGE AR 6-34
 - SET FILE-CACHE-TIMER-INTERVAL AR 6-35
 - SET FILE-INFO-FLUSH-AGE AR 6-36
 - SET FILE-INFO-IDLE-FLUSH-AGE AR 6-37
 - SET MAXIMUM-CACHE-BUFFERS AR 6-38
 - SET MAXIMUM-CACHE-FILES AR 6-39
 - SET MAXIMUM-DIRTY-BUFFERS AR 6-40
 - SET MAXIMUM-FILESYSTEM-BUFFERS AR 6-41
 - SET MAXIMUM-FILESYSTEM-CHANNELS AR 6-42
 - SET MAXIMUM-FILESYSTEM-FILES AR 6-43
 - SET MAXIMUM-OPEN-CHANNELS AR 6-44
 - SET MAXIMUM-QUEUED-REMOVES AR 6-45
 - SET MAXIMUM-WRITE-JOBS AR 6-46
 - SET NUMBER-OF-DUPLICATE-REQUESTS-CACHED AR 6-47
 - SET NUMBER-OF-RPC-TRANSPORTS AR 6-48
 - SET READ-ONLY-FLUSH-AGE AR 6-49
 - SET READ-WRITE-FLUSH-AGE AR 6-50
 - SET SECONDS-BEFORE-WRITEBACK AR 6-51
 - SET USE-DIRECTORY-BLOCKING-ASTS AR 6-52
 - SET USE-FILE-BLOCKING-ASTS AR 6-53
 - SHOW AR 6-54

- SPAWN AR 6-55
- STATUS AR 6-57
- USE AR 6-58
- VERSION AR 6-59
- WRITE AR 6-60
- NLST/LIST commands AD 11-1
- MULTINET RMTALLOC AR 1-80
- DNS
 - NSLOOKUP command
 - set AD 6-33
- node names, resolving DN 2-4
- NOFDL_FILES AD 20-16
- NOLINKS AD 20-16
- nomodify AD 7-20
- nopeer AD 7-21
- nopwd AD 21-9
- noquery AD 7-20
- noserve AD 7-20
- NOSTREAM_CONVERSION AD 20-16
- NOT-CONFIG command
 - ADD NAME-MAPPING DN A-3
 - ADD OBJECT DN A-4
 - ADD PROXY DN A-6
 - ATTACH DN A-8
 - DELETE NAME DN A-9
 - DELETE OBJECT DN A-10
 - DELETE PROXY DN A-11
 - EXIT DN A-12
 - GET DN A-13
 - HELP DN A-14
 - NETCONTROL DN A-15
 - PUSH DN A-16
 - QUIT DN A-17
 - RELOAD DN A-18
 - SAVE DN A-19
 - SET DN A-20
 - SHOW DN A-21
 - SPAWN DN A-23
 - STATUS DN A-25
 - USE DN A-26
 - VERSION DN A-27
 - WRITE DN A-28
- notrust AD 7-21
- NOUNIQUE_FILENO AD 20-16
- NOVERSIONS AD 20-17
- NOVMS_ACCESS_CHECKING AD 20-17
- NSLOOKUP command
 - exit AR 1-68
 - finger AR 1-68
 - help AR 1-68
 - ls name AR 1-69
 - lserver AR 1-69
 - name AR 1-68
 - name server AR 1-68
 - root AR 1-69
 - set all AR 1-68
 - set class AR 1-68
 - set d2 AR 1-68
 - set debug AR 1-68
 - set defname AR 1-68
 - set domain AR 1-68
 - set port AR 1-68
 - set query-type AR 1-68
 - set recurse AR 1-68
 - set retry AR 1-68
 - set root AR 1-68
 - set srchlist AR 1-68
 - set timeout AR 1-68
 - set type AR 1-68
 - set vc AR 1-68
- NSLOOKUP command server AR 1-68
- NSLOOKUP, using to debug DNS AD 6-33
- NSLOOKUP/TYPE qualifier AR 1-70
- NTP
 - access control commands AD 7-20
 - configuration
 - commands AD 7-14
 - configuring AD 7-9
 - files AD 7-11
 - functions AD 7-9
 - managing AD 7-9
 - miscellaneous command AD 7-21
 - monitoring commands AD 7-17
- NTP.CONF AD 7-11
 - converting to AD 7-12
- NTP.DRIFT AD 7-11
- NTP.KEYS AD 7-11
- NTPDATE utility AD 7-34
- ntpport AD 7-21
- NTPQ utility AD 7-25
- NTPSERVER.LOG AD 7-11
- NTPTRACE utility AD 7-35
- NTY devices AD 4-23
- NTYCP command
 - CREATE PORT AR 7-3
 - DELETE PORT AR 7-5
 - EXIT AR 7-6
 - HELP AR 7-7
 - MODIFY PORT AR 7-8
- NTYCP CREATE PORT /LOGICAL keyword option
 - MODE AR 7-4, AR 7-9
 - NAME AR 7-4, AR 7-8
 - TABLE AR 7-4, AR 7-9
- NTYSMB symbiont AD 9-19

O

- online help AD 1-6
- opcom AD 7-16
- OpenVMS
 - Access Control Lists (ACLs)

- controlling NFS file access with AD 19-22
- channel usage parameters AD 19-42
- channels AD 19-39
- text files to UNIX text files, mapping AD 19-9
- user accounts for client users, creating AD 19-12
- OpenVMS error values ME 1-3
- OpenVMS mail, using across the network UG 3-1
- operation
 - cancel or close AD 4-16
- optional zone statements AD 6-15
- OSI reference model IN 7-5

P

- packet
 - filter file AD 3-22
 - filtering for security AD 3-21
- PAK (Product Authorization Key) IN 1-9, DN 1-2
- parameters
 - DNSKEYGEN
 - n AR 1-16
- passphrase UG 8-20, UG 8-22
 - forgotten UG 8-22
 - lost UG 8-22
- password authentication AD 21-9
- password-based authentication AD 21-2
- passwords AD 3-29
- patterns
 - host name AD 21-7
 - hostname AD 21-14, AD 21-16
 - hostnames AD 21-18
 - port number AD 21-14, AD 21-16
 - rights identifier AD 21-7
 - user name AD 21-7
- PC-NFSD remote printing service AD 19-28
- peer AD 7-14
 - hosts, determining AD 7-13
- peerstats AD 7-18
- permit list AD 12-21
- PIN AD 4-36
- PING IN 6-3
- Pipelining (RFC-2197) AD 8-2
- pool permit lists AD 12-21
- POP
 - logical names AD 8-22
 - using MULTINET_POPx_FLAGS logical AD 8-22
- port forwarding
 - definition UG 8-8
- Post Office Protocol (POP) AD 8-21
- PPP (Point-to-Point Protocol)
 - configuration parameters AD 3-43
- PRESERVE_DATES AD 20-17
- print queue
 - parameters, adding AD 9-11
 - troubleshooting AD 9-21
- print queues, configuring AD 9-6
- PRINTER AD 19-30
- printer queues AD 9-14
- PRINTER-CONFIG command
 - ADD AR 9-5
 - ATTACH AR 9-6
 - CLEAR AR 9-8
 - DELETE AR 9-9
 - ERASE AR 9-10
 - EXIT AR 9-11
 - GET AR 9-12
 - HELP AR 9-13
 - MODIFY AR 9-14
 - PUSH AR 9-15
 - QUIT AR 9-16
 - SAVE AR 9-17
 - SELECT AR 9-18
 - SET ALLOW-USER-SPECIFIED-PRINTER AR 9-19
 - SET BASE-PRIORITY AR 9-20
 - SET BLOCK-LIMIT-LOWER AR 9-21
 - SET BLOCK-LIMIT-UPPER AR 9-22
 - SET BURST AR 9-23
 - SET CHARACTERISTICS AR 9-24
 - SET DEFAULT-FORM AR 9-25
 - SET DESCRIPTION AR 9-26
 - SET FLAG AR 9-27
 - SET LIBRARY AR 9-28
 - SET NOFEED AR 9-29
 - SET OWNER AR 9-30
 - SET PROTECTION AR 9-31
 - SET RETAIN-ON-ERROR AR 9-32
 - SET SCHEDULE-NOSIZE AR 9-33
 - SET SEPARATE-BURST AR 9-34
 - SET SEPARATE-FLAG AR 9-35
 - SET SEPARATE-RESET AR 9-36
 - SET SEPARATE-TRAILER AR 9-37
 - SET SUPPRESS-EQJ-FF AR 9-38
 - SET SUPPRESS-REMOTE-BANNER AR 9-39
 - SET SUPPRESS-TELNET AR 9-40
 - SET TAB-EXPAND AR 9-41
 - SET TRAILER AR 9-42
 - SET WS-DEFAULT AR 9-43
 - SET WS-EXTENT AR 9-44
 - SET WS-QUOTA AR 9-45
 - SHOW AR 9-46
 - SPAWN AR 9-47
 - STATUS AR 9-49
 - VERSION AR 9-51
 - WRITE AR 9-52
- printers on remote systems AD 9-4
- process memory AD 19-38
- Process Software World Wide Web server AD 1-6
- protocol definitions AD 6-4
- proxies, using DN 1-3
- pseudo device interface (pd) AD 3-20
- pseudoterminal AD 21-12

PSI service parameters AD 3-37
public-key cryptography UG 8-2
PUSH AR 10-16
pwdlifetime AD 21-22

Q

QIO interface IN 7-3

QIO interface call

IO\$_ACCEPT PR 4-2
IO\$_ACCEPT_WAIT PR 4-4
IO\$_BIND PR 4-5
IO\$_CONNECT PR 4-6
IO\$_GETPEERNAME PR 4-7
IO\$_GETSOCKNAME PR 4-8
IO\$_GETSOCKOPT PR 4-9
IO\$_IOCTL PR 4-11
IO\$_LISTEN PR 4-12
IO\$_READVBLK PR 4-13
IO\$_RECEIVE PR 4-13
IO\$_SELECT PR 4-15
IO\$_SEND PR 4-17
IO\$_SENSEMODE PR 4-19
IO\$_SENSEMODE|IO\$_M_CTRL PR 4-22
IO\$_SETMODE|IO\$_M_ATTNAST PR 4-32
IO\$_SETSOCKOPT PR 4-33
IO\$_SHUTDOWN PR 4-35
IO\$_SOCKET PR 4-36
SYS\$CANCEL PR 4-38
SYS\$DASSGN PR 4-39

qualifiers

ADD MOUNT-RESTRICTION
-ro AR 6-8

CHECK

IGNORE_ERRORS AR 1-6
OUTPUT AR 1-6
VERBOSE AR 1-6

CONFIGURE

ACCESS AR 1-7
CONFIGURATION_FILE AR 1-9
DECNET AR 1-7
INTERFACES AR 1-7
MAIL AR 1-7
MENU AR 1-7
NETWARE AR 1-8
NETWORK AR 1-8
NFS AR 1-8
NOBOLD AR 1-9
NOT AR 1-8
PRINTERS AR 1-8
SERVER_IMAGE AR 1-9
SERVERS AR 1-8

CREATE PORT

LOG AR 7-3
LOGICAL AR 7-3

NODE AR 7-4
PORT AR 7-4
SERVICE AR 7-4

DCL

CREATE_NTU UG 5-8
ESCAPE_CHARACTER UG 5-6

DECwindows

NODE UG 7-1
TRANSPORT UG 7-1

DELETE PORT

LOG AR 7-5

DIG

ADDITIONAL AR 1-10
ADDRESS AR 1-10
ANSWER AR 1-10
AUTHORITY AR 1-11
CLASS AR 1-11
CMD AR 1-11
DEBUG AR 1-11
DEBUG2 AR 1-11
ENVSAVE AR 1-11
ENVSET AR 1-11
FILE AR 1-12
HEADER AR 1-12
HFLAGS AR 1-12
IGNORE AR 1-12
KEEPOPEN AR 1-12
KEY AR 1-12
NOADDITIONAL AR 1-10
NOANSWER AR 1-10
NOAUTHORITY AR 1-11
NOCMD AR 1-11
NODEBUG AR 1-11
NODEBUG2 AR 1-11
NOHEADER AR 1-12
NOHFLAGS AR 1-12
NOIGNORE AR 1-12
NOKEEPOPEN AR 1-12
NOPFDEF AR 1-12
NOPFMIN AR 1-13
NOQUERY AR 1-13
NOQUESTION AR 1-13
NORECURSE AR 1-13
NOREPLY AR 1-13
NOSTATS AR 1-14
NOSTICKY AR 1-14
NOVC AR 1-14
PFAND AR 1-12
PFDEF AR 1-12
PFMIN AR 1-13
PFOR AR 1-13
PFSET AR 1-13
PING AR 1-13
PORT AR 1-13
QUERY AR 1-13
QUESTION AR 1-13

RECURSE AR 1-13
 REPLY AR 1-13
 RETRY AR 1-13
 SERVER AR 1-14
 STATS AR 1-14
 STICKY AR 1-14
 TIMEOUT AR 1-14
 TIMEWAIT AR 1-14
 TYPE AR 1-14
 VC AR 1-14
 DNSKEYGEN
 -a AR 1-17
 -c AR 1-17
 -D AR 1-16
 DSA_DSS AR 1-16
 -F AR 1-17
 -h AR 1-17
 HOST_KEY AR 1-17
 LARGE_EXPONENT AR 1-17
 NOAUTHENTICATION AR 1-17
 NOENCRYPTION AR 1-17
 -p AR 1-17
 PROTOCOL AR 1-17
 -R AR 1-16
 RSA AR 1-16
 -s AR 1-17
 STRENGTH AR 1-17
 -u AR 1-17
 USER_KEY AR 1-17
 -z AR 1-17
 ZONE_KEY AR 1-17
 DNSSIGNER
 BIND AR 1-20
 -bind AR 1-20
 DEBUG AR 1-21
 -dur AR 1-23
 DURATION AR 1-23
 -ess AR 1-23
 -k1 AR 1-24
 KEY AR 1-24
 -ks AR 1-24
 -l AR 1-21
 -n AR 1-21
 NXT AR 1-21
 -or AR 1-21
 ORIGIN AR 1-21
 PARENT AR 1-21
 POLICY AR 1-22
 -pt AR 1-24
 PURGE_PERIOD AR 1-24
 SELF_SIGN AR 1-23
 SIG AR 1-23
 -st AR 1-24
 STATISTICS AR 1-24
 ZONE AR 1-24
 FINGER
 CLUSTER UG A-4
 NOCLUSTER UG 2-3, UG A-4
 FONT COMPILE
 BIT_ORDER AR 1-26
 BYTE_ORDER AR 1-26
 OUTPUT AR 1-26
 PADDING AR 1-26
 SCANLINE AR 1-26
 SERVER AR 1-26
 FONT INFO
 OUTPUT AR 1-27
 SERVER AR 1-27
 FONT LIST
 BOUNDS AR 1-28
 COLUMNS AR 1-28
 LISTING_TYPE AR 1-28
 NOSORT AR 1-28
 OUTPUT AR 1-28
 SERVER AR 1-28
 WIDTH AR 1-28
 FONT SHOW
 BIT_ORDER AR 1-30
 BITMAP_PADDING AR 1-30
 BYTE_ORDER AR 1-30
 END AR 1-30
 EXTENTS AR 1-30
 OUTPUT AR 1-30
 PADDING AR 1-30
 SCANLINE AR 1-31
 SERVER AR 1-31
 START AR 1-31
 FONT UNCOMPILE
 OUTPUT AR 1-32
 SERVER AR 1-32
 FTP
 ACCOUNT UG A-5
 BINARY UG A-5
 FDL UG 6-7, UG 6-8
 IMAGE UG A-5
 INITIALIZATION UG A-6
 MODE UG A-6
 NOINITIALIZATION UG 6-16
 NONPASV UG 6-15
 NOVMS_STRUCTURE_NEGOTIATION
 UG A-7
 PASSWORD UG A-6
 PASV UG 6-15
 PASV DCL UG 6-15
 PASV=NEGOTIATE UG 6-15
 PORT UG A-6
 PROMPT UG A-6
 STATISTICS UG A-6
 STRUCTURE UG A-6
 TAKE_FILE UG A-7
 TYPE UG A-7
 TYPE=EBCDIC UG 6-8

USERNAME UG A-7
 VERBOSE UG A-7
 VMS_STRUCTURE_NEGOTIATION UG A-7
 WINDOW_SIZE UG 6-6

GET
 FDL UG B-33

HOST_TABLE COMPILE
 HOST_TABLE_FILE AR 1-33
 SILENTLY AR 1-33
 STARTING_HASH_VALUE AR 1-33
 TBLUK_FILE AR 1-33
 UNIX_HOST_FILE AR 1-34

HOST_TABLE GET
 HOST AR 1-35
 OUTPUT_FILE AR 1-35
 QUERY AR 1-35
 SILENTLY AR 1-35
 VERSION AR 1-35

KERBEROS
 AUTH UG 4-2, UG 4-3
 AUTHENTICATION=KERBEROS UG 4-3
 CHECK_TGT UG 4-3
 REALM UG 4-2
 USERNAME UG 4-2, UG 4-3

KERBEROS DATABASE DUMP
 DATABASE_FILE AR 1-38

KERBEROS DATABASE EDIT
 DATABASE_FILE AR 1-39
 PROMPT_FOR_KEY AR 1-39

KERBEROS DATABASE INITIALIZE
 DATABASE_FILE AR 1-42
 REALM AR 1-42

KERBEROS DATABASE LOAD
 DATABASE_FILE AR 1-43

KERBEROS DATABASE SRVTAB
 PROMPT AR 1-45
 REALM AR 1-45

KERBEROS DESTROY
 QUIET UG A-9
 STATUS UG A-9

KERBEROS INIT
 INSTANCE UG A-10
 LIFETIME UG A-10
 REALM UG A-10
 USERNAME UG A-10
 VERBOSE UG A-10

KERBEROS LIST
 BRIEF UG A-11
 CHECK_TGT UG A-11
 SRVTAB UG A-11

KERBEROS PASSWORD
 INSTANCE UG A-12
 REALM UG A-12
 USERNAME UG A-12

LPRM
 ALL UG A-13
 NODE UG A-13

 QUEUE UG A-13
 SUPERUSER UG A-13
 USER UG A-13

MODIFY PORT
 LOG AR 7-8
 LOGICAL AR 7-8
 NODE AR 7-9
 PORT AR 7-9
 SERVICE AR 7-9

NETCONTROL
 HOST AR 1-48
 VERBOSE AR 1-48

NFSDISMOUNT
 ALL AR 1-61
 LOG AR 1-61

NFSMOUNT
 FID_CACHE AR 1-62
 LOCKING AR 1-62
 PAGEFILE AR 1-62
 PORT AR 1-63
 PRIORITY AR 1-63
 PROCESSOR AR 1-63
 READ_SIZE AR 1-63
 RELOAD AR 1-63
 SEMANTICS AR 1-63
 SOFT AR 1-66
 TIMEOUT AR 1-66
 TRANSPORT AR 1-66
 UNIQUE_FILENO AR 1-66
 VMS_SERVER AR 1-66
 VOLUME AR 1-66
 WRITE AR 1-66
 WRITE_SIZE AR 1-66
 WSEXTENT AR 1-66
 WSQUOTA AR 1-67

NSLOOKUP
 CLASS AR 1-69
 DEBUG AR 1-69
 DEBUG2 AR 1-69
 DEFNAMES AR 1-69
 DNSRCH AR 1-70
 DOMAIN AR 1-70
 IGNTC AR 1-70
 NODEBUG AR 1-69
 NODEBUG2 AR 1-69
 NODEFNAMES AR 1-69
 NODNSRCH AR 1-70
 NOIGNTC AR 1-70
 NORECURSE AR 1-70
 NOVC AR 1-71
 PORT AR 1-70
 RECURSE AR 1-70
 RETRY AR 1-70
 ROOT_SERVER AR 1-70
 TIMEOUT AR 1-70
 TYPE AR 1-70

VC AR 1-71
NSLOOKUP/TYPE
A AR 1-70
ANY AR 1-70
AXFR AR 1-70
CNAME AR 1-71
GID AR 1-71
HINFO AR 1-71
MAILB AR 1-71
MB AR 1-71
MG AR 1-71
MINFO AR 1-71
MR AR 1-70
MX AR 1-70
NS AR 1-70
PTR AR 1-71
SOA AR 1-71
TXT AR 1-71
UID AR 1-71
UINFO AR 1-71
WKS AR 1-71
NSUPDATE
-d AR 1-73
DEBUG AR 1-73
-K AR 1-73
KEY AR 1-73
NODEBUG AR 1-73
NOVC AR 1-73
-V AR 1-73
VC AR 1-73
PING
DATA_LENGTH AR 1-75
DEBUG AR 1-75
FLOOD AR 1-75
NUMBER_OF_PACKETS AR 1-76
PRELOAD AR 1-76
QUIET AR 1-76
RECORD_ROUTE AR 1-76
ROUTE AR 1-76
VERBOSE AR 1-76
PROFILE/DELETE
CONFIRM AR 2-3
LOG AR 2-3
PROFILE/MODIFY
CONFIRM AR 2-4
LOG AR 2-4
PROFILE/SHOW
FULL AR 2-5
PUT
FDL UG B-62
RCP
AUTHENTICATION=KERBEROS UG A-14
EXACT UG A-15
LOG UG A-15
PASSWORD UG 6-2, UG A-15
RECURSIVE UG A-14, UG A-15
TRUNCATE_USERNAME UG A-15
USERNAME UG 6-2, UG A-15
VMS_ATTRIBUTES UG A-16
RDATE
DELTA AR 1-77
LOG AR 1-77
SET AR 1-77
RECEIVE
FDL UG B-67
RLOGIN
AUTHENTICATION=KERBEROS UG A-20
BUFFER_SIZE UG A-20
DEBUG UG A-20
EIGHT_BIT UG A-20
PORT UG A-20
TRUNCATE_USERNAME UG A-20
USERNAME UG A-21
RMTALLOC
CD AR 1-78
HEADER AR 1-85
HOST AR 1-85
LOG AR 1-78
PASSWORD AR 1-79
SEMANTICS AR 1-79
TRUNCATE_USERNAME AR 1-81
UNIX_SERVER AR 1-81
USERNAME AR 1-81
VMS_ATTRIBUTES AR 1-81
WRITE AR 1-81
RSHELL
ERROR UG 5-2, UG A-22
INPUT UG 5-2, UG A-22
INPUT=NLA0 UG 5-2
OUTPUT UG 5-2, UG A-22
PASSWORD UG 5-2, UG A-22, UG A-23
PORT UG A-22
TRUNCATE_USERNAME UG A-23
USERNAME UG 5-2, UG A-23
RUSERS
ALL UG A-24
FULL UG A-24
NOALL UG A-24
NOFULL UG A-24
SAVE
STARTUP AR 5-28
SEND
AND_MAIL UG A-25
FDL UG B-75
FOREIGN UG A-3
OR_MAIL UG A-25
SET
ABORT_OUTPUT_CHARACTER UG C-21
ACCOUNT UG B-76
APPROXIMATE_TEXT_SIZE AR 6-32
ARE_YOU_THERE_CHARACTER UG C-22
AUTH UG C-33

- AUTOFLUSH UG C-23
- BREAK_CHARACTER UG C-24
- DEBUG UG C-25
- ERASE_CHARACTER_CHARACTER
 - UG C-26
- ERASE_LINE_CHARACTER UG C-27
- ESCAPE_CHARACTER UG C-28
- INTERRUPT_PROCESS_CHARACTER
 - UG C-30
- LOCAL_FLOW_CONTROL UG C-31
- LOG_FILE UG C-32
- PASSWORD UG B-76
- UNIX UG C-34
- USER UG B-76
- SET /ARP
 - ADD AR 1-86
 - COMMUNITY_NAME AR 1-86
 - DELETE AR 1-86
 - FLUSH AR 1-86
 - PERMANENT AR 1-86
 - PROXY AR 1-86
 - PUBLISH AR 1-86
 - SNMP_HOST AR 1-86
- SET /DECNET
 - BUFFERS AR 1-88
 - CLOSE AR 1-88
 - CONNECT AR 1-88
 - DEVICE AR 1-88
 - FILTER_OUT_OF_ORDER AR 1-88
 - LOGDATA AR 1-88
 - LOGERRORS AR 1-88
 - PORT AR 1-88
 - REMOTE_ADDRESS AR 1-88
 - TCP AR 1-89
- SET /INTERFACE
 - ADDRESS AR 1-90
 - ARP AR 1-90
 - COMMUNITY_NAME AR 1-91
 - D1 AR 1-91
 - D2 AR 1-91
 - D3 AR 1-91
 - DEBUG AR 1-91
 - DECNET_ETHERNET_ADDRESS AR 1-91
 - DOWN AR 1-91
 - DYNAMIC AR 1-91
 - FFI_BUFFERS AR 1-91
 - FILTER AR 1-92
 - HARDWARE_DEVICE AR 1-92
 - IP_BROADCAST AR 1-92
 - IP_SUBNET_MASK AR 1-92
 - LINK_LEVEL AR 1-92
 - LOCAL AR 1-93
 - MTU AR 1-93
 - MULTICAST AR 1-93
 - PEER AR 1-93
 - POINT_TO_POINT_DESTINATION AR 1-93
 - PPP_NOICMP AR 1-93
 - PPP_OPTIONS AR 1-93
 - PROTOCOL AR 1-93
 - RARP AR 1-93
 - SNMP_HOST AR 1-94
 - TRAILERS AR 1-94
 - UP AR 1-94
 - VMS_DEVICE AR 1-94
- SET /ROUTE
 - ADD AR 1-95
 - COMMUNITY_NAME AR 1-95
 - DELETE AR 1-95
 - FLUSH AR 1-95
 - FORCE_HOST AR 1-95
 - FORCE_NETWORK AR 1-95
 - NETWORK_IMAGE AR 1-96
 - SNMP_HOST AR 1-96
- SET /TIMEZONE
 - FILES AR 1-97
 - LOG AR 1-97
 - SELECT AR 1-97
- SHOW
 - ALL AR 1-98
 - ARP AR 1-98
 - BUFFERS AR 1-98
 - COMMUNITY_NAME AR 1-98
 - CONFIGURATION AR 1-98
 - CONNECTIONS AR 1-98
 - CONTINUOUS AR 1-98
 - FULL AR 1-99, AR 6-54, AR 10-57
 - HOSTS AR 1-99
 - INTERFACE AR 1-99
 - IP AR 1-99
 - LICENSE AR 1-99
 - MIB_VAR AR 1-99
 - NFSMOUNT AR 1-99
 - OUTPUT AR 1-100
 - PROTOCOLS AR 1-100
 - QUEUE AR 1-100
 - REMOTE_HOST AR 1-100
 - ROUTE AR 1-100
 - RPC_PORTMAP AR 1-101
 - SNMP_HOST AR 1-101
 - STATISTICS AR 1-101
 - SYMBOLIC_ADDRESSES AR 1-101
 - TCP AR 1-101
 - VERSION AR 1-101
 - WIDTH AR 1-102
- SKEY
 - COUNT AR 2-7
 - DELETE AR 2-7
 - OUTPUT AR 2-7
 - PRINT AR 2-7
 - QUEUE AR 2-7
 - SYMBOL AR 2-7

- SPAWN
 - INPUT UG B-79, UG C-35, AR 3-17,
AR 4-43, AR 5-53, AR 6-55,
AR 8-20, AR 9-47, AR 10-59
 - LOGICAL_NAMES UG B-79, UG C-35,
AR 3-17, AR 4-43, AR 5-53,
AR 6-55, AR 8-20, AR 9-47,
AR 10-59
 - OUTPUT UG B-79, UG C-35, AR 3-17,
AR 4-43, AR 5-53, AR 6-55,
AR 8-20, AR 9-47, AR 10-59
 - SYMBOLS UG B-79, UG C-35, AR 3-17,
AR 4-43, AR 5-53, AR 6-55,
AR 8-20, AR 9-47, AR 10-59
 - WAIT UG B-79, UG C-35, AR 3-17, AR 4-43,
AR 5-53, AR 6-55, AR 8-20,
AR 9-47, AR 10-59
- TALK
 - OLD UG A-26
- TCPDUMP
 - AFTER AR 1-106
 - BEFORE AR 1-106
 - COUNT AR 1-106
 - DEBUG AR 1-106
 - DEVICE AR 1-106
 - DOMAINS AR 1-106
 - EBCDIC AR 1-106
 - ETHERNET_HEADER AR 1-106
 - FOREIGN_NUMERICALLY AR 1-106
 - HEXADECIMAL_DUMP AR 1-106
 - INTERFACE AR 1-106
 - NUMERICALLY AR 1-106
 - OUTPUT AR 1-107
 - QUIET AR 1-107
 - READ_BINARY AR 1-107
 - RPC AR 1-107
 - SNAPSHOT_SIZE AR 1-107
 - TIMESTAMPS AR 1-107
 - VERBOSE AR 1-107
 - WRITE_BINARY AR 1-107
- TCPVIEW
 - COUNT AR 1-109
 - DEVICE AR 1-109
 - DOMAINS AR 1-109
 - ETHERNET_HEADER AR 1-109
 - FILE_FORMAT AR 1-109
 - INTERFACE AR 1-109
 - PROMISCUOUS AR 1-110
 - SNAPSHOT_SIZE AR 1-110
 - TIMESTAMPS AR 1-110
 - VERBOSE AR 1-110
- TELNET
 - ABORT_OUTPUT_CHARACTER UG A-28
 - ARE_YOU_THERE_CHARACTER UG A-28
 - AUTHENTICATION=KERBEROS UG A-28
 - AUTOFLUSH UG A-28
 - BREAK_CHARACTER UG A-28
 - BUFFER_SIZE UG A-29
 - CREATE_NTY UG A-29
 - DEBUG UG A-30
 - DELETE_NTY UG A-30
 - ERASE_CHARACTER_CHARACTER
UG A-30
 - ERASE_LINE_CHARACTER UG A-30
 - ESCAPE_CHARACTER UG A-30
 - INTERRUPT_PROCESS_CHARACTER
UG A-31
 - LOCAL_FLOW_CONTROL UG A-31
 - LOG_FILE UG A-31
 - PORT UG A-31
 - PRINT_ESCAPE_CHARACTER UG A-32
 - PROTOCOL UG A-32
 - TCP UG A-32
 - TERMINAL_TYPE UG A-32, UG C-37
 - TN3270=AUTOMATIC UG A-32
 - TN5250 UG 5-9
 - TN5250=AUTOMATIC UG A-32
 - UNIX UG A-32
 - VERSION UG A-33
- TN3270
 - YALE UG 5-19
- TOKEN CRYPTOCARD/CLEAR
 - LOG AR 2-9
- TOKEN CRYPTOCARD/LOAD
 - CHALLENGE AR 2-10
 - CONFIRM AR 2-10
 - DISPLAY AR 2-10
 - KEY AR 2-11
 - LANGUAGE AR 2-12
 - LOG AR 2-12
 - PIN AR 2-12
 - TIMEOUT AR 2-13
 - VERBOSE AR 2-13
- TOKEN SKEY/CLEAR
 - LOG AR 2-20
- TOKEN SKEY/INITIALIZE
 - LOG AR 2-21
 - PASSWORD AR 2-21
 - SEED AR 2-22
 - SEQUENCE AR 2-22
 - VERBOSE AR 2-22
- TOKEN SNK/CLEAR
 - LOG AR 2-26
- TOKEN SNK/LOAD
 - CONFIRM AR 2-27
 - KEY AR 2-27
 - LOG AR 2-27
 - MODE AR 2-27
 - VERBOSE AR 2-28
- TRACEROUTE
 - DEBUG AR 1-115
 - MAXIMUM_TTL AR 1-115

- MINIMUM_TTL AR 1-115
 - NUMBER_OF_PROBES AR 1-115
 - OUTPUT AR 1-115
 - PORT AR 1-115
 - ROUTE AR 1-115
 - SOURCE AR 1-115
 - SYMBOLIC_ADDRESSES AR 1-115
 - TYPE_OF_SERVICE AR 1-115
 - VERBOSE AR 1-115
 - WAIT_TIME AR 1-116
 - WHOIS
 - HOST UG A-35
 - OUTPUT UG A-35
 - PORT UG A-35
 - X11DEBUG
 - LOG AR 1-117
 - query programs AD 7-9
 - queue groups AD 8-11
-
- R**
- R services
 - authentication UG 5-3
 - configuring AD 4-19
 - RLOGIN AD 4-19
 - RSHELL AD 4-19
 - random number generator AD 21-20
 - RARP (Reverse Address Resolution Protocol) AD 12-2
 - clients AD 12-3
 - configuration
 - file AD 12-4
 - reloading AD 12-4
 - packet reception, enabling AD 12-3
 - service AD 12-3
 - rawstats AD 7-18
 - RCP
 - requirements UG 6-1
 - using UG 6-1, UG 6-2
 - using Kerberos with UG 4-3
 - recurse AD 6-33
 - recv() PR 2-4, PR 2-5
 - recvfrom() PR 2-4, PR 2-5
 - regenerate server key AD 21-8
 - release notes, printing IN 2-3
 - remote
 - login, controlling AD 10-8
 - magnetic tape server, configuring AD 10-1
 - printer queues, checking AD 9-5
 - remote host information AD 15-9
 - remote hosts, specifying UG 2-1
 - remote login program
 - first authentication method UG 8-1
 - fourth authentication method UG 8-4
 - second authentication method UG 8-1
 - third authentication method UG 8-2
 - REPLY_TO header AD 8-10
 - requestkey AD 7-16
 - retry timers AD 9-10
 - REWIND AR 1-80
 - REXEC AD 4-22
 - RFC
 - 1001 AD 12-57
 - 1001/1002 AD 12-56
 - 1002 AD 12-57
 - 1032 AD 6-8
 - 1033 AD 6-8
 - 1034 AD 6-8
 - 1035 AD 6-8, AD 6-9, AD 12-54, AD 12-55, AD 12-57
 - 1042 AD 12-55
 - 1084 AD 12-4
 - 1105 AD 5-3
 - 1112 AD 6-6
 - 1122 AD 12-53, AD 12-58
 - 1123 AD 6-6
 - 1179 AD 12-56
 - 1191 AD 12-57
 - 1256 AD 12-58
 - 1869 AD 8-2
 - 2131 AD 12-13
 - 2132 AD 12-13
 - 2197 AD 8-2
 - 2741 AD 15-9
 - 2742 AD 15-9
 - 827 AD 5-3
 - 865 AD 12-6, AD 12-53
 - 868 AD 12-60
 - 887 AD 12-58
 - 888 AD 5-3
 - 891 AD 5-3
 - 893 AD 12-60
 - 894 AD 12-55
 - 904 AD 5-3
 - 911 AD 5-3
 - 950 AD 5-3, AD 12-59
 - 951 AD 12-4
 - 952 AD 6-3, AD 6-6
 - RFC (Requests for Comment) IN 8-1
 - RHOSTS UG 5-4, AD 4-19, AD 4-20, AD 4-21, AD 4-22, AD 10-8
 - rhhosts authentication AD 21-2, AD 21-19
 - RhostsAuthentication UG 8-14
 - RhostsRSAAuthentication UG 8-14
 - rights identifier patterns AD 21-7
 - RIP protocol
 - configuring AD 5-9
 - RLOGIN AD 4-19, AD 6-2, AD 16-11, AD 21-2
 - and RSHELL authentication cache AD 4-21
 - terminating UG 5-3
 - using UG 5-2
 - using Kerberos with UG 4-3

RMS-E-PRV, insufficient privilege or file protection violation ME 2-23

RMT

- client AD 10-4
- configuration AD 10-1
- server tape drive name qualifiers AD 10-3

RMTALLOC AD 10-5

- CD-ROM access AD 10-6
- qualifiers, using AD 10-7
- tape drive access AD 10-5

router discovery IN 6-10, AD 5-2

- service parameters AD 5-3

routing

- definition IN 6-9
- table IN 6-10

RSA authentication UG 8-20

RSA authentication identity UG 8-22, UG 8-24

RSA challenge-response authentication AD 21-2

RSA host authentication AD 21-2, AD 21-10

RSA key

- bits AD 21-13
- comment AD 21-13
- exponent AD 21-13
- modulus AD 21-13
- options AD 21-13

RSA key file

- Allowforwardingport AD 21-14
- Allowforwardingto AD 21-14
- command AD 21-15
- Denyforwardingport AD 21-15
- Denyforwardingto AD 21-16
- from AD 21-16
- idle-timeout AD 21-17
- no-agent-forwarding AD 21-17
- no-port-forwarding AD 21-17
- no-X11-forwarding AD 21-17

RSA key file examples AD 21-18

RSA keys AD 21-13

RSA-based authentication UG 8-2

RSA-based host authentication UG 8-1

RSHELL AD 4-19, AD 4-21, AD 21-2

- connection, disabling the standard error AD 4-21
- executing commands UG 5-1
- using UG 5-1
- using Kerberos with UG 4-3

S

S/KEY authentication AD 4-38

S/KEY clients, unpacking IN 1-19

sa_data PR 2-1

sa_family PR 2-1

Safe-failover boot file AD 12-69

SAVE AR 11-12

search list AD 6-12

secure encrypted communications AD 21-1

secure shell

- configuration file
- keyword
- BatchMode UG 8-11
- Cipher UG 8-11
- ClearAllForwardings UG 8-11
- Compression UG 8-11
- CompressionLevel UG 8-11
- ConnectionAttempts UG 8-11
- EscapeChar UG 8-11
- FallBackToRsh UG 8-12
- ForwardAgent UG 8-12
- ForwardX11 UG 8-12
- GatewayPorts UG 8-12
- GlobalKnownHostsFile UG 8-12
- Host UG 8-12
- IdentityFile UG 8-12
- KeepAlive UG 8-13
- LocalForward UG 8-13
- NumberOfPasswordPrompts UG 8-13
- PasswordAuthentication UG 8-13
- PasswordPromptHost UG 8-13
- PasswordPromptLogins UG 8-13
- Port UG 8-13
- ProxyCommand UG 8-14
- RemoteForward UG 8-14
- RhostsAuthentication UG 8-14
- RhostsRSAAuthentication UG 8-14
- RSAAAuthentication UG 8-14
- StrictHostKeyChecking UG 8-15
- UsePrivilegedPort UG 8-15
- UserKnownHostsFile UG 8-15
- UseRsh UG 8-15
- configuration files UG 8-10

Secure Shell (SSH)

- daemon (SSHD) AD 21-1
- preparations before running IN 1-23
- restrictions AD 21-1
- security AD 21-2
- server AD 21-1
- understanding AD 21-1

SSHD AD 21-2

SSHD_MASTER AD 21-1

secure shell client UG 8-1

Secure/IP

- client logical names AD 4-30
- terminology
- Authentication AD 3-34
- Cardcode AD 3-34
- DES AD 3-34
- Method AD 3-34
- Passcode AD 3-34
- PIN AD 3-34
- Plaintext passwords AD 3-34

- Seed AD 3-35
- Sequence AD 3-35
- TLN (Trusted Local Network) AD 3-35
- Token AD 3-35
- SECUREIP_CONFIGURE.COM AD 4-25
- SecureNet key authentication AD 4-37
- SecurID authentication AD 4-35
- SECURID_CLIENT_CHECK, using IN 1-18
- security AD 19-3
 - and file protections AD 20-4
- Security Dynamics SecurID card AD 3-31
- SELECT AR 10-20
- select list AD 4-16
- select() PR 3-1
- send() PR 2-4, PR 2-5
- sendto() PR 2-4, PR 2-5
- SERVER AD 19-30
- server AD 7-14
 - access AD 4-10
- server key AD 21-3
- server listens AD 21-4
- SERVER-CONFIG AD 4-2
 - commands AD 4-3
 - service parameters AD A-1
 - services provided with MultiNet AD A-3
 - utility AD A-1
- SERVER-CONFIG command
 - ADD AR 10-5
 - ATTACH AR 10-6
 - COPY AR 10-8
 - DISABLE AR 10-10
 - ENABLE AR 10-11
 - EXIT AR 10-12
 - GET AR 10-13
 - NETCONTROL AR 10-15
 - QUIT AR 10-17
 - RESTART AR 10-18
 - SAVE AR 10-19
 - SET ACCEPT-HOSTS AR 10-21
 - SET ACCEPT-NETS AR 10-22
 - SET BACKLOG AR 10-23
 - SET CONNECTED AR 10-24
 - SET DISABLED-NODES AR 10-25
 - SET ENABLED-NODES AR 10-26
 - SET FLAGS AR 10-27
 - SET INIT AR 10-30
 - SET KEEPALIVE-TIMERS AR 10-31
 - SET LISTEN AR 10-32
 - SET LOG-ACCEPTS AR 10-33
 - SET LOG-FILE AR 10-34
 - SET LOG-REJECTS AR 10-35
 - SET MAX-SERVERS AR 10-36
 - SET PARAMETERS AR 10-37
 - SET PRIORITY AR 10-38
 - SET PROCESS AR 10-39
 - SET PROGRAM AR 10-40
 - SET RECEIVE-BUFFER-SPACE AR 10-42
 - SET REJECT-BY-DEFAULT AR 10-41
 - SET REJECT-HOSTS AR 10-43
 - SET REJECT-MESSAGE AR 10-44
 - SET REJECT-NETS AR 10-45
 - SET SEND-BUFFER-SPACE AR 10-46
 - SET SERVICE AR 10-47
 - SET SERVICE-NAME AR 10-48
 - SET SERVICE-TYPE AR 10-49
 - SET SOCKET-FAMILY AR 10-50
 - SET SOCKET-OPTIONS AR 10-51
 - SET SOCKET-PORT AR 10-52
 - SET SOCKET-TYPE AR 10-53
 - SET USERNAME AR 10-54
 - SET WORKING-SET AR 10-55
 - SET WORKING-SET-QUOTA AR 10-56
- SHOW AR 10-57
- SHUTDOWN AR 10-58
- SPAWN AR 10-59
- STATUS AR 10-61
- USE AR 10-62
- VERSION AR 10-63
- WRITE AR 10-64
- servers and clients AD 19-2
- service
 - configuration AD 4-2
 - definitions AD 6-4
 - ratings setting AD 6-35
- service configuration information IN 1-22
- SIGURG PR 4-1
- sin_addr PR 2-2, PR 2-5
- sin_family PR 2-2, PR 2-6
- sin_len PR 2-6
- sin_port PR 2-2, PR 2-5
- sin_zero PR 2-2
- slewalways AD 7-15
- SLIP
 - configuration parameters AD 3-40
 - understanding AD 3-38
- SMTP
 - configuration file AD 8-1
 - host aliases, specifying AD 8-15
 - queues AD 6-8
 - service for ALL-IN-1 users, configuring AD 8-24
 - symbiont AD 6-8
 - configuring AD 8-9
- SMTP/MR
 - configuration, completing AD 8-30
 - document conversion, configuring AD 8-29
- SMTP/MR, configuring AD 8-25
- SMTP-DECnet mail gateway, configuring AD 8-31
- SMTP-HOST-NAMES AD 8-15
- SMTP-to-DECnet mail AD 8-32
- SMUX peers, adding AD 15-8
- SNMP AD 15-1
 - (Simple Network Management Protocol) IN 6-13

- agent AD 15-2
- agent extensibility AD 15-9
- communities IN 6-14
- managers, agents, and trap sinks AD 15-1
- multiplexing (SMUX) protocol AD 15-8
- services
 - community parameters AD 15-6
 - configuration file AD 15-4
 - configuring AD 15-2
 - extendible MIBs, supporting AD 15-9
 - log file AD 15-9
 - management objects, defining values AD 15-4
- traps
 - clientcontrollable AD 15-8
 - enabling/disabling AD 15-8
 - troubleshooting AD 15-9
- traps IN 6-13
- SNMP-CONFIG command
 - ATTACH AR 11-4
 - CHECK AR 11-5
 - EXIT AR 11-6
 - HELP AR 11-7
 - INITIALIZE AR 11-8
 - PUSH AR 11-9
 - QUIT AR 11-10
 - RELOAD AR 11-11
 - SET AUTH-TRAPS AR 11-13
 - SET DHCP-COLDSTART AR 11-14
 - SET DHCP-SHUTDOWN AR 11-15
 - SET DNS-COLDSTART AR 11-16
 - SET DNS-SHUTDOWN AR 11-17
 - SET MASTER-AGENT-MAXMSG AR 11-18
 - SET MASTER-AGENT-PORT AR 11-19
 - SET READ-COMMUNITY AR 11-20
 - SET SNMP-MAXMSG AR 11-21
 - SET SYSCONTACT AR 11-22
 - SET SYSDSCR AR 11-23
 - SET SYSLOCATION AR 11-24
 - SET TRAP-COMMUNITY AR 11-25
 - SET TRAP-DESTINATIONS AR 11-26
 - SET WRITE-COMMUNITY AR 11-27
 - SHOW AR 11-28
 - SPAWN AR 11-29
 - USE AR 11-30
 - VERSION AR 11-31
 - WRITE AR 11-32
- SNMPD.CONF file AD 15-4
- SOCK_STREAM PR 2-2
- sockaddr PR 2-1, PR 2-2, PR 2-5
- sockaddr_in PR 2-1, PR 2-2, PR 2-3, PR 2-4, PR 2-5
- socket definition PR 2-1
- socket library function
 - accept() PR 3-3
 - bcmp() PR 3-5
 - bcopy() PR 3-6
 - bind() PR 3-7
 - bzero() PR 3-8
 - connect() PR 3-9
 - endhostent() PR 3-11
 - endnetent() PR 3-12
 - endprotoent() PR 3-13
 - endservent() PR 3-14
 - getdtablesize() PR 3-15
 - gethostbyaddr() PR 3-16
 - gethostbyname() PR 3-18
 - gethostbynameaddr() PR 3-19
 - gethostname() PR 3-20
 - getnetbyaddr() PR 3-21
 - getnetbyname() PR 3-22
 - getpeername() PR 3-23
 - getprotobyname() PR 3-24
 - getprotobynumber() PR 3-25
 - getprotoent() PR 3-26
 - getservbyname() PR 3-27
 - getservbyport() PR 3-28
 - getservent() PR 3-29
 - getsockname() PR 3-30
 - getsockopt() PR 3-31
 - gettimeofday() PR 3-33
 - hostalias() PR 3-34
 - htonl() PR 3-35
 - htons() PR 3-36
 - inet_addr() PR 3-37
 - inet_lnaof() PR 3-38
 - inet_makeaddr() PR 3-39
 - inet_netof() PR 3-40
 - inet_network() PR 3-41
 - inet_ntoa() PR 3-42
 - klread() PR 3-43
 - klseek() PR 3-44
 - klwrite() PR 3-45
 - listen() PR 3-46
 - multinet_kernel_nliith PR 3-47
 - nlist() PR 3-48
 - ntohl() PR 3-49
 - ntohs() PR 3-50
 - recv() PR 3-51
 - recvfrom() PR 3-53
 - recvmsg() PR 3-55
 - select() PR 3-57
 - select_wake() PR 3-60
 - send() PR 3-61
 - sendmsg() PR 3-62
 - sendto() PR 3-64
 - sethostent() PR 3-66
 - setnetent() PR 3-67
 - setprotoent() PR 3-68
 - setservent() PR 3-69
 - setsockopt() PR 3-70
 - shutdown() PR 3-72
 - socket ioctl

- FIONBIO PR 3-77
- FIONREAD PR 3-78
- SIOCADDRT PR 3-79
- SIOCATMARK PR 3-83
- SIOCDAEP PR 3-84
- SIOCDELRT PR 3-81
- SIOCGARP PR 3-85
- SIOCGIFADDR PR 3-87
- SIOCGIFBRDADDR PR 3-89
- SIOCGIFCONF PR 3-91
- SIOCGIFDSTADDR PR 3-92
- SIOCGIFFLAGS PR 3-94
- SIOCGIFMETRIC PR 3-96
- SIOCGIFNETMASK PR 3-98
- SIOCSARP PR 3-86
- SIOCSIFADDR PR 3-88
- SIOCSIFBRDADDR PR 3-90
- SIOCSIFDSTADDR PR 3-93
- SIOCSIFFLAGS PR 3-95
- SIOCSIFMETRIC PR 3-97
- SIOCSIFNETMASK PR 3-99
- socket option
 - SO_BROADCAST PR 3-100
 - SO_DEBUG PR 3-101
 - SO_DONTROUTE PR 3-102
 - SO_ERROR PR 3-103
 - SO_KEEPAIVE PR 3-104
 - SO_LINGER PR 3-105
 - SO_OOBINLINE PR 3-106
 - SO_RCVBUF PR 3-107
 - SO_RCVLOWAT PR 3-108
 - SO_RCVTIMEO PR 3-109
 - SO_REUSEADDR PR 3-110
 - SO_SNDBUF PR 3-111
 - SO_SNDLOWAT PR 3-112
 - SO_SNDTIMEO PR 3-113
 - SO_TYPE PR 3-114
 - TCP_KEEPAIVE PR 3-115
 - TCP_NODELAY PR 3-116
- socket() PR 3-73
- socket_close() PR 3-75
- socket_ioctl() PR 3-76
- socket_perror() PR 3-117
- socket_read() PR 3-118
- socket_write() PR 3-119
- vms_errno_string() PR 3-120
- socket() PR 2-2, PR 4-1
- socket_read() PR 2-3, PR 2-4
- socket_write() PR 2-3, PR 2-4
- software patches AD 1-7
- SPAWN DN A-23
- spoofing
 - DNS UG 8-1
 - IP UG 8-1
 - routing UG 8-1
- SSH
 - authentication agent UG 8-20
 - command options UG 8-6
 - daemon files AD 21-23
 - SSHD.LOG AD 21-23
 - SSHD_MASTER.LOG AD 21-23
 - START_SSH.COM AD 21-23
 - logicals AD 21-21
 - MULTINET_SSH_ALLOW_EXPIRED_PW AD 21-22
 - MULTINET_SSH_ALLOW_PREEXPIRED_PW AD 21-23
 - MULTINET_SSH_KEYGEN_MIN_PW_LEN AD 21-23
 - MULTINET_SSH_PARAMETERS AD 21-23
 - MULTINET_SSH_USE_SYSGEN_LGI AD 21-23
 - SSH_DIR AD 21-21
 - SSH_EXE AD 21-22
 - SSH_LOG AD 21-22
 - SSH_MAX_SESSIONS AD 21-22
 - SSH_TERM_MBX AD 21-22
 - starting the server AD 21-11
- SSH command
 - ALLOW_REMOTE_CONNECT UG 8-6
 - CIPHER UG 8-6
 - COMPRESSION UG 8-6
 - DEBUG UG 8-6
 - ESCAPE_CHARACTER UG 8-7
 - IDENTITY_FILE UG 8-7
 - LOCAL_FORWARD UG 8-7
 - LOG_FILE UG 8-7
 - NO_AGENT_FORWARDING UG 8-7
 - OPTION UG 8-7
 - PORT UG 8-7
 - QUIET UG 8-8
 - REMOTE_FORWARD UG 8-8
 - USE_NONPRIV_PORT UG 8-8
 - USERNAME UG 8-8
 - VERSION UG 8-8
- SSH command line option
 - bits AD 21-3
 - config_file AD 21-3
 - debug AD 21-3
 - host AD 21-3
 - host_key_file AD 21-3
 - key_gen_time AD 21-3
 - login_grace_time AD 21-3
 - port AD 21-4
 - quiet_mode AD 21-4
 - version AD 21-4
- SSH files
 - AUTHORIZED_KEYS UG 8-16
 - CONFIG. UG 8-16
 - HOSTS.EQUIV UG 8-16
 - IDENTITY. UG 8-16
 - IDENTITY.PUB UG 8-16

- KNOWN_HOSTS UG 8-17
- RANDOM_SEED. UG 8-17
- RHOSTS UG 8-18
- SHOSTS UG 8-18
- SHOSTS.EQUIV UG 8-19
- SSH_CONFIG UG 8-19
- SSH_KNOWN_HOSTS UG 8-19
- SSH_KNOWN_HOSTS
 - file
 - AUTHORIZED_KEYS AD 21-21
 - MULTINET
 - HOSTS.EQUIV AD 21-19
 - SHOSTS.EQUIV AD 21-19
 - SSH_HOST_KEY AD 21-19
 - SSH_HOST_KEY.PUB AD 21-20
 - SSH_KNOWN_HOSTS AD 21-20
 - SSH_RANDOM_SEED AD 21-20
 - SSHD_CONFIG AD 21-21
 - SHOSTS AD 21-21
 - SY\$LOGIN
 - RHOSTS AD 21-21
- SSHADD UG 8-20, UG 8-21
- SSHADD option
 - DELETE UG 8-21
 - LIST UG 8-21
 - PURGE UG 8-21
- SSHAGENT UG 8-20
 - authentication agent UG 8-21
 - authentication private keys UG 8-20
- SSHD AD 21-2, AD 21-12
- SSHD configuration file keyword
 - AllowForwardingPort AD 21-4
 - AllowForwardingTo AD 21-5
 - AllowGroups AD 21-5
 - AllowHosts AD 21-5
 - AllowSHosts AD 21-6
 - AllowTcpForwarding AD 21-6
 - AllowUsers AD 21-6
 - DenyForwardingPort AD 21-6
 - DenyForwardingTo AD 21-7
 - DenyGroups AD 21-7
 - DenyHost AD 21-7
 - DenySHosts AD 21-7
 - DenyUsers AD 21-7
 - FascistLogging AD 21-7
 - ForcedEmptyPasswdChange AD 21-7
 - HostKey AD 21-7
 - IdleTimeout AD 21-8
 - IgnoreRhosts AD 21-8
 - KeepAlive AD 21-8
 - KeyRegenerationInterval AD 21-8
 - ListenAddressee AD 21-8
 - LoginGraceTime AD 21-8
 - PasswordAuthentication AD 21-9
 - PermitEmptyPasswords AD 21-9
 - PermitRootLogin AD 21-9
 - Port AD 21-9
 - QuietMode AD 21-9
 - RandomSeed AD 21-9
 - RhostsAuthentication AD 21-10
 - RhostsRSAAuthentication AD 21-10
 - RSAAuthentication AD 21-10
 - ServerKeyBits AD 21-10
 - SilentDeny AD 21-10
 - StrictModes AD 21-10
 - SyslogFacility AD 21-10
 - X11DisplayOffset AD 21-10
 - X11ForwardingSM AD 21-10
- SSHD_MASTER AD 21-1, AD 21-12
- SSHKEYGEN UG 8-22
 - authentication key pairing UG 8-22
 - definition UG 8-22
 - file
 - IDENTITY UG 8-24
 - IDENTITY.PUB UG 8-24
 - RANDOM_SEED UG 8-24
 - option
 - BITS UG 8-23
 - CHANGE_CIPHER UG 8-23
 - CHANGE_COMMENT UG 8-23
 - CHANGE_PASSPHRASE UG 8-23
 - COMMENT UG 8-23
 - HOST UG 8-23
 - IDENTITY_FILE UG 8-23
 - NEW_PASSPHRASE UG 8-23
 - PASSPHRASE UG 8-23
- STAT command AD 11-6
- static
 - IP routes, configuring AD 5-2
 - SLIP interfaces, configuring AD 3-39
- static leases AD 12-66
- stats AD 7-16
- stolen key AD 21-16
- STREAM
 - protocol queue, configuring AD 9-6
 - queues, advantages over AD 9-19
- subclass declaration AD 12-46
- Sun
 - host clients, configuring AD 19-20
 - microsystems hosts, booting diskless AD 19-33
- supported network interface devices AD 3-3
- SYLOGIN.COM, inhibiting output from UG 6-3
- SYMBIONT AD 19-30
- symbiont file
 - MULTINET_LPD_SYMBIONT.EXE IN 3-1
 - MULTINET_NTYSMB.EXE IN 3-1
 - MULTINET_NW_PRINT_SYMBIONT.EXE IN 3-1
 - MULTINET_SMTP_SYMBIONT.EXE IN 3-1
 - MULTINET_STREAM_SYMBIONT.EXE IN 3-1
- synchronized
 - hosts AD 7-10
 - timekeeping AD 7-9

SYS\$LOGIN

- .RHOSTS UG 5-3, AD 4-20, AD 10-1

SYSLOG AD 4-41

- message classes AD 4-42

- system startup command procedure, modifying IN 1-21

- system startup, modifying DN 2-2

- system/vendor information

- limiting AD 8-9

T

TCP

- client PR 2-3

- program PR A-1

- server PR 2-4

- programs PR B-1

TCP/IP

- concepts IN 6-3

- broadcast addresses IN 6-5

- host names IN 6-5

- IP addresses IN 6-3

- LAN (Local Area Network) hardware

- addresses IN 6-3

- operation IN 6-5

- physical networks IN 6-3

- subnet masks IN 6-4

- networking IN 6-1

- protocols IN 6-6

- IP (Internet Protocol) IN 6-6

- PPP (Point-to-Point Protocol) IN 6-8

- SLIP (Serial Line Internet Protocol) IN 6-8

- TCP (Transmission Control Protocol) IN 6-7

- UDP (User Datagram Protocol) IN 6-8

- TCP/IP connections AD 21-2

- TCP/IP transport over UCX AD 3-48

- TCPDUMP IN 6-3, AD 6-2, AD 16-11, AR 1-105,
AR 1-109

- TCPVIEW IN 6-3, AR 1-109

- technical support AD 1-3

- TELNET AD 6-2

- command

- ABORT UG C-4

- ATTACH UG C-5

- ATTN UG C-6

- AYT UG C-7

- BINARY UG C-8

- BREAK UG C-9

- BYE UG C-10

- CLOSE UG C-11

- CONNECT UG C-12

- CREATE-NTY UG C-13

- DEBUG UG C-14

- ECHO UG C-15

- EXIT UG C-16

- HELP UG C-17

- LOG-FILE UG C-18

- PUSH UG C-19

- QUIT UG C-20

- SET ABORT-OUTPUT-CHARACTER
UG C-21

- SET ARE-YOU-THERE-CHARACTER
UG C-22

- SET AUTO-FLUSH UG C-23

- SET BREAK-CHARACTER UG C-24

- SET DEBUG UG C-25

- SET ERASE-CHARACTER-

- CHARACTER UG C-26

- SET ERASE-LINE-CHARACTER UG C-27

- SET ESCAPE-CHARACTER UG C-28

- SET EXTENDED UG C-29

- SET INTERRUPT-PROCESS-

- CHARACTER UG C-30

- SET LOCAL-FLOW-CONTROL UG C-31

- SET LOG-FILE UG C-32

- SET REMOTE-USERNAME UG C-33

- SET UNIX-LINE-TERMINATOR UG C-34

- SPAWN UG C-35

- STATUS UG C-36

- TERMINAL-TYPE UG C-37

- VERSION UG C-38

- commands, using UG 5-5

- control sequence

- ABORT-OUTPUT UG 5-7

- ARE-YOU-THERE UG 5-7

- BREAK-CHARACTER UG 5-7

- ERASE-CHARACTER UG 5-8

- ERASE-LINE UG 5-8

- INTERRUPT-PROCESS UG 5-8

- control sequences, using UG 5-7

- logging in with UG 5-5

- server AD 4-23

- starting UG 5-5

- troubleshooting UG 5-21

- using Kerberos with UG 4-3

- TELNET sessions UG 8-8

- TERMINAL AD 19-30

TFTP

- command

- CONNECT UG D-2

- GET UG D-3

- PUT UG D-4

- QUIT UG D-5

- REXMT UG D-6

- STATUS UG D-7

- TIMEOUT UG D-8

- TRACE UG D-9

- copying files using UG 6-17

- requirements UG 6-17

- using UG 6-17

- TFTP (Trivial File Transfer Protocol) AD 4-18

- ticket status, checking UG 4-3

tickets, acquiring and deleting UG 4-2
 timezone
 parameters AD 19-46
 support AD 7-2
 timezone configuration AD 7-1
 TN3270 AD 2-8
 application keypad access UG 5-19
 emulation UG 5-19
 function key mapping UG 5-14
 translation table mapping UG 5-19
 using transparent mode UG 5-18
 TN5250 AD 2-8
 application keypad access UG 5-19
 TN5250 function key mapping UG 5-16
 tokens AD 3-30
 TRACEROUTE IN 6-3
 traps AD 15-2
 trusted local network AD 4-27
 trustedkey AD 7-16
 tunneling UG 8-8
 typographical conventions UG 1-2, PR 1-2, AD 1-2

U

UCX-compatible services AD 4-16
 UDP PR 2-4
 client program PR C-1
 server programs PR D-1
 UIC protection AD 19-25
 UID/GID mappings AD 19-14, AD 20-3, AD 20-9
 UNIX
 /etc/hosts file, converting AD 6-6
 bootptab file AD 12-10
 device special files AD 19-7
 devices AD 10-5
 file
 links AD 19-6
 names AD 11-8, AD 19-7
 system semantics AD 19-6
 setuid, setgid, and "sticky" file modes AD 19-7
 style listings AD 11-6
 UNIX errno message values ME 1-3
 UNLOAD AR 1-80
 unsecure connections UG 8-8
 untrusted hosts UG 8-1
 UPPER_CASE_DEFAULT AD 20-17
 USE AR 9-50
 user
 equivalences UG 5-3
 information, displaying UG 2-2
 user exits, adding and updating IN 1-22
 user name patterns AD 21-7
 user profiles, deleting AD 4-33
 user-specified print

destinations AD 9-12
 parameters AD 9-13
 utility
 PHONE UG 2-4
 RLOGIN UG 5-1
 RSHELL UG 5-1
 TALK UG 2-4
 TELNET UG 5-1

V

vc AD 6-33
 vendor encapsulated options AD 12-63
 verbose logging AD 21-7
 virtual and physical memory AD 19-38
 VMS
 MAIL AD 8-17
 VMS_FILENAMES AD 20-17
 VMS_SERVER AD 20-17
 VMS_STYLE_CREATE
 mount point option AD 19-28
 VMScluster aliasing AD 3-49
 VMSINSTAL, running IN 1-10
 VMS-to-VMS negotiation AD 10-7
 VRFY AD 8-10

W

WHOIS UG A-35, AD 3-2, AR 5-49
 write protection AD 10-9
 writeback cache parameters AD 19-44

X

X display management AD 13-1
 X11 connections AD 21-2
 X11 forwarding AD 21-10
 X11DEBUG IN 6-3
 X11-Gateway
 concepts AD 17-1
 debugging AD 17-7
 error messages AD 17-7
 security AD 17-6
 X11R3 displays AD 13-9
 Xauthority data UG 8-5
 XDM
 administrative tasks AD 13-3
 configuration, reloading AD 13-8
 server AD 13-2
 configuration AD 13-4
 controlling the AD 13-6
 resources AD 13-4

Index

XDM.SERVERS file AD 13-10

XDMCP requests AD 13-2

XNTPDC utility AD 7-28

Z

zone types AD 6-17

Reader's Comments

MultiNet for OpenVMS User's Guide, V. 4.3 Part Number: N-5010-43-NN-A

Your comments and suggestions will help us to improve the quality of our future documentation. Please note that this form is for comments on documentation only.

I rate this guide's:	Excellent	Good	Fair	Poor
Accuracy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Completeness (enough information)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Clarity (easy to understand)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Organization (structure of subject matter)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Figures (useful)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Index (ability to find topic)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ease of use	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

1. I would like to see more/less: _____
2. Does this guide provide the information you need to perform daily tasks? _____
3. What I like best about this guide: _____
4. What I like least about this guide: _____
5. Do you like this guide's binding? If not, what would you prefer? _____

My additional comments or suggestions for improving this guide:

I found the following errors in this guide:

Page	Description
------	-------------

_____	_____
_____	_____

Please indicate the type of user/reader that you most nearly represent:

System Manager	<input type="radio"/>	Educator/Trainer	<input type="radio"/>
Experienced Programmer	<input type="radio"/>	Sales	<input type="radio"/>
Novice Programmer	<input type="radio"/>	Scientist/Engineer	<input type="radio"/>
Computer Operator	<input type="radio"/>	Software Support	<input type="radio"/>
Administrative Support	<input type="radio"/>	Other (please specify)	<input type="radio"/> _____

Name: _____ Dept. _____
Company: _____ Date _____
Mailing Address: _____

After filling out this form, FAX or mail it to:

Process Software, 959 Concord Street, Framingham, MA 01701-4682
Attention: Technical Publications Group FAX 508-879-0042 e-mail: techpubs@process.com

